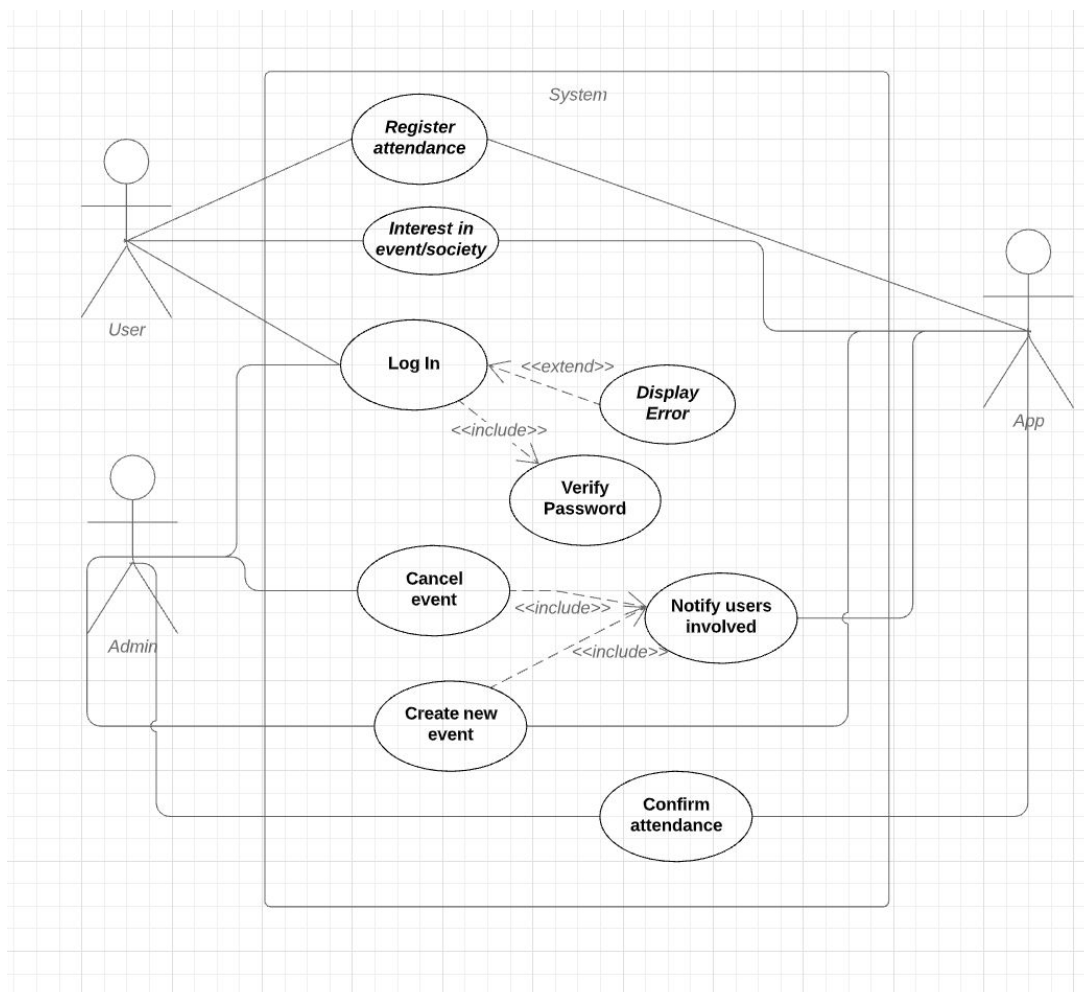This UML case diagram describes the process our minimum viable product should contain. After the meetings with our client we came to the conclusion that these features were the most important and should be the ones initially implemented to reach the goals of the project.

There are two types of users in our app, a regular user (students) and admins. Both of these types of users will have to sign in when using the app and our app will always check if the password is correct. If the password is incorrect, they will be notified as shown on the case diagram with the <<extend>> arrow.

The main functionality the regular users are meant to have is the ability to register interest in societies and sports events as well as registering their attendance if they have attended a specific event. The main functionality of the admin is the ability to create and cancel events while being able to notify all interested users about a new event or a cancelled event. The <<include>> tag specifies that each time an event is created or cancelled, the admin will automatically notify everyone by mass email or a notification on the app. They are also able to confirm attendance of the students to send to the sports center.

The UML diagram shows the classes we intend to use to complete the MVP. We can see a User class which is the parent class of the Student and the Admin class, which are the only types of users using the app. Each user has a login ID as well as a password which will be checked during the login.

We use composition to describe the Event and Society class in relation to the Admin. Events and societies cannot exist without the admin creating them initially and performing administrative roles such as creating or cancelling events. This is also true for users with their interested events as well as the respective societies they follow.

We chose to use this model as it shows the core functionality of the app, everything implemented after this will rely on this main functionality so it is important that we implement it properly. Furthermore, it is the main functionality asked by the client to be implemented.