The system was tested according to the requirements that the group wrote in order to assure the system’s functionality. The GUI was tested as well to ensure that it follows what was specified in the Use Case descrpitons.

GUI

1. Store event
2. The user first chooses the type of event he wishes to add
3. The user enters the required data into the system
4. The system verifies the data and stores it

PASSED

1. Search events
2. The user enters a search parameter
3. The system checks and lists every event that meets the requirements
4. If there is no event with that requirement, the list displayed will be empty

PASSED

1. Modify event
2. The user selects the desired event
3. The user enters new data
4. The system verifies the data and stores it

PASSED

1. Delete event
2. The user selects the event that needs to be deleted
3. The system deletes the event

PASSED

1. Store member information
2. The user enters the required data into the system
3. The system verifies the data and stores it

PASSED

1. Search members
2. The user enters a search parameter
3. The system checks and lists every member that meets the requirements
4. If there is no member with that requirement, the list displayed will be empty

PASSED

1. Modify member information
2. The user selects the desired member
3. The user enters new data
4. The system verifies the data and stores it

PASSED

1. Delete member
2. The user selects the member whose information needs to be deleted
3. The system deletes the member’s information

PASSED

1. Store lecturer information
2. The user enters the required data into the system
3. The system verifies the data and stores it

PASSED

1. Search lecturers
2. The user enters a search parameter
3. The system checks and lists every lecturer that meets the requirements
4. If there is no lecturer with that requirement, the list displayed will be empty

PASSED

1. Modify lecturer information
2. The user selects the desired lecturer
3. The user enters new data
4. The system verifies the data and stores it

PASSED

1. Delete lecturer
2. The user selects the lecturer whose information needs to be deleted
3. The system deletes the lecturer’s information

PASSED

SYSTEM

1. The user can create a new event PASSED
2. Events can be of four different types PASSED
3. The system should store about lectures: a title, start date, start time, duration, lecturer, 1 subject, sponsor name, price, finalized or not, total number of tickets, discount PASSED
4. The system should store about seminars: a title, start date, start time, duration, lecturers, subjects, sponsor name, price, finalized or not, total number of tickets, discount PASSED
5. The system should store about workshops: a title, start date, start time, duration, lecturers, food included (vegan or not), price, finalized or not, total number of tickets, discount PASSED
6. The system should store about trips: a title, start date, start time, duration, locations, price, finalized or not, total number of tickets, discount PASSED
7. The user should be able to search events by: finalized or not, start date, subject, price,lecturers, sponsors PASSED
8. The user should be able to modify every aspect an event PASSED
9. The user can store a member’s information (name, email, address, phone, payment year, date of registration, newsletter subscription, attended events) PASSED
10. The user should be able to search members by name, payment year, date of registration, attended events PASSED
11. The user should be able to update the information of each member PASSED
12. The user can store a lecturer’s information (name, email, phone, sponsor or not, subject) PASSED
13. The user should be able to search lecturers by name, subject, email, phone, sponsor or not PASSED
14. The user should be able to update the information of each lecturer PASSED

Besides using the final version of the system to test its functionality there has been created a tester class in order to check for more specific errors.

 

 