**REQUIREMENTS**

Vipassanā – Insight Awareness (VIA) - System

**Students**

Stefan Harabagiu – 266116

Andrei Cioanca – 266105

Dawei Li – 269053

Nikita Roshkov - 266900

**Supervisors**

Mona Wendel Andersen

Henrik Kronborg Pedersen

Michael Viuff

1. **Functional requirements**
2. The user should be able to create a new event
3. The events can be of four different types (lectures, seminars, workshops and trips)
4. The system should store the following information about the lectures: a title, start date, start time, duration, lecturer, 1 subject, sponsor name, price, finalized or not, total number of tickets, discount
5. The system should store the following information about the seminars: a title, start date, start time, duration, lecturers, subjects, sponsor name, price, finalized or not, total number of tickets, discount
6. The system should store the following information about the workshops: a title, start date, start time, duration, lecturers, food included (vegan or not), price, finalized or not, total number of tickets, discount
7. The system should store the following information about the trips: a title, start date, start time, duration, locations, price, finalized or not, total number of tickets, discount
8. The user should be able to search events by: finalized or not, start date, subject, price,lecturers, sponsors
9. The user should be able to modify every aspect of an event
10. The user should be able to store members’ information
11. Members are defined by name, email, address, phone, payment year, date of registration, newsletter subscription, attended events
12. The user should be able to search members by name, payment year, date of registration, attended events
13. The user should be able to update the information of each member
14. The user should be able to store lecturers’ information
15. Lecturers are defined by name, email, phone, sponsor or not, subject
16. The user should be able to search lecturers by name, subject, email, phone, sponsor or not
17. The user should be able to update the information of each lecturer
18. **Non-Functional requirements**
19. The system has to be implemented in Java
20. The system has to be compatible with Microsoft Windows 7,8,10
21. The system needs to be able to run indefinitely
22. The system should use files for secondary storage only

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Requirement | Use Case reference | Test reference |
| 1 | Functional.1 | Store event | Test.SYSTEM.1 |
| 2 | Functional.2 | Store event | Test.SYSTEM.2 |
| 3 | Functional.3 | Store event | Test.SYSTEM.3 |
| 4 | Functional.4 | Store event | Test.SYSTEM.4 |
| 5 | Functional.5 | Store event | Test.SYSTEM.5 |
| 6 | Functional.6 | Store event | Test.SYSTEM.6 |
| 7 | Functional.7 | Search event | Test.SYSTEM.7 |
| 8 | Functional.8 | Edit & Delete event | Test.SYSTEM.8 |
| 9 | Functional.9 & Functional.10 | Store member | Test.SYSTEM.9 |
| 10 | Functional.11 | Search member | Test.SYSTEM.10 |
| 11 | Functional.12 | Edit member | Test.SYSTEM.11 |
| 12 | Functional.13 & Functional.14 | Store lecturer | Test.SYSTEM.12 |
| 13 | Functional.15 | Search lecturer | Test.SYSTEM.13 |
| 14 | Functional.16 | Edit lecturer | Test.SYSTEM.14 |