

VIPASSANA MANAGEMENT SYSTEM – VIA

SEP1 project report

Supervisors:

Mona Wendel Andersen

Michael Viuff

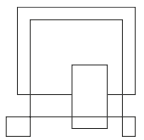
19 / 12 / 2017

Version history

Description	Version	Init	Revised	Approved	Date
Start of document	0.1.0	IHA	<INIT>	<INIT>	2017-01-31
Small corrections	0.2.0	IHA	MWA		2017-02-01
Revisions made in workshop meeting	1.0.0	MWA/IHA	All	ICT	2017-08-19

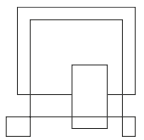
Group 3

Student Name	Student No
Angel Petrov	266489
Diyar Hussein	266352
Rodrigo Pereira	269084
Rytis Kunigiskis	266109



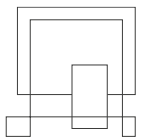
Contents

1	Introduction	3
2	Requirements	4
1.	Functional Requirements	4
2.	Non-Functional Requirements	4
3	Analysis	5
4	Design.....	9
5	Implementation.....	13
6	Test	15
7.	Results and Discussion	17
8	Conclusion	18
9	Project future	19
10	References	20
11	List of Appendixes.....	21



Abstract

The objectives of the project are to construct a system for the client company – Vipassana Insight Awareness (VIA) that will perform the requirements that clients request and which will address the way that the company operates and how it works with information management. The technical choices behind constructing the system are that it must follow a set of predefined requirements, must be able to function, given those requirements and must satisfy the needs of the clients. The results of this are a system that not only provides proper functionality, but also, passes almost all testing criteria's and can be put into use.



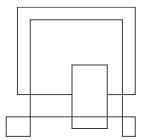
1 Introduction

Vipassanā - Insight Awareness (VIA) is a centre for spiritual events originally with a base in the Buddhist principles of meditation as an insight with awareness of what is happening as it happens. They organize spiritual events, send out newsletters and sometimes articles to their members.

Vipassanā does not store any trace of lecturers, members, plans for events, work done by its council or whom they promised advertisements free of charge. This gives the rise of a potential loss of clients which in turn opens a liability in the way that the organisation operates, its potential achievements and overall stability.

Studies have proven that Vipassanā's meditation is rewarding to the individual - benefits such as improvement of psychological health caused the means to solve the problem in the current situation of Vipassanā, the project will benefit a wide audience of people (*Szeker, R. A., and Wertheim, E. H.*). Furthermore, the end results for an individual who undergoes one of the company's offered courses yields an increase in positivity, balance, cooperativeness and improvements in mindfulness. (*Front Psychol*).

The company has provided key information at the interview in regards to improving their internal data management system and how the company would like to work with and manage information in a way so that it can keep track of members, events, lecturers and work which is done by its council in a simplistic way so that they are able to improve their business strategy prosperity and efficiency whilst addressing their current liability in how the organisation operates.



2 Requirements

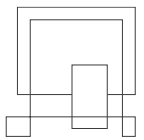
Requirements concerning system practicality and functionality are as follows:

1. Functional Requirements

- 1 The user should be able to create, modify and remove events within the system.
- 2 The system should be able to store the customer's name, phone number, payment year, date of membership and e-mail address.
- 3 The system should be able to list all events.
- 4 The system should provide the user the possibility to leave feedback on events on behalf of customers.
- 5 The user should be able to book events within the system.
- 6 The system should have a member list that includes name, phone number, date of membership and year of payment, and optionally an e-mail address.
- 7 The user should be able to search for lecturers and events within the system.
- 8 The user should be able to search in a specific category within the system.
- 9 The user should be able to modify member and lecturer information within the system.
- 10 The user should be able to list all members within the system.
- 11 The user should be able to add new members and lecturers within the system.
- 12 The user should be able to finalize events within the system.
- 13 The system should be able to display feedback from events to sponsors.

2. Non-Functional Requirements

- 1 The system needs to answer within 3 seconds 95% of the time.
- 2 The system has to be implemented in Java.
- 3 The system development process must fit in within the customer's budget and needs.
- 4 The system should be able to read and save to secondary storage based on the Java loose coupling.



3 Analysis

3.1 Use Case Diagram

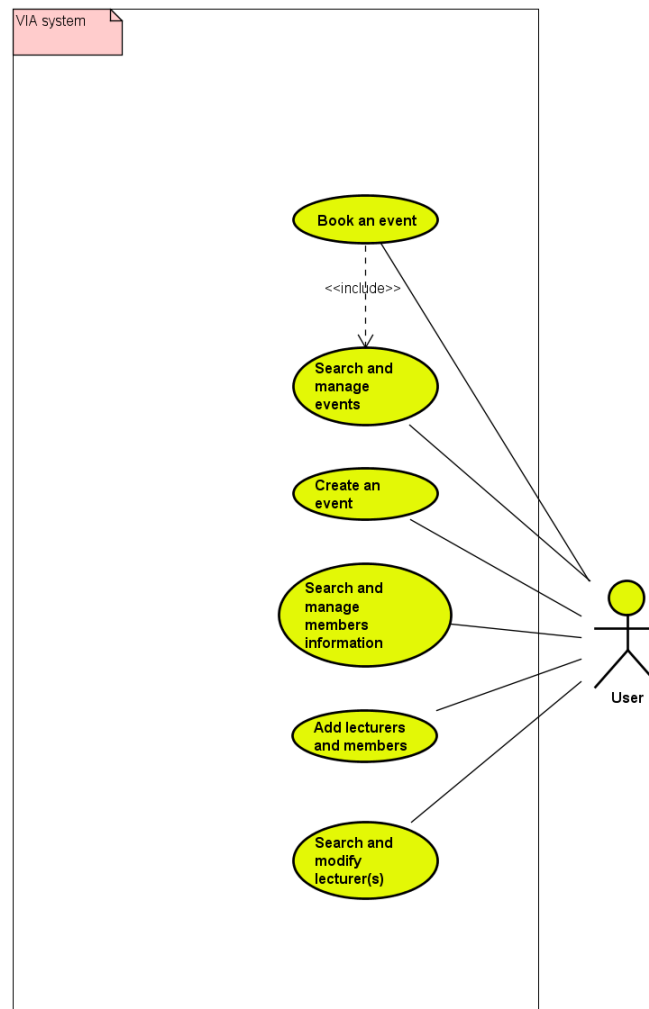
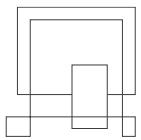


Figure 1 – Use Case diagram

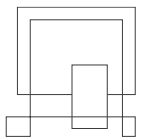
The figure above shows the Use Case diagram which represents system functionalities of the program. Within the diagram, the user has the possibility to make use of the corresponding Use Cases. The “User” can be any staff personnel such as “employee” or “staff” that the client company associates with the use of the system.



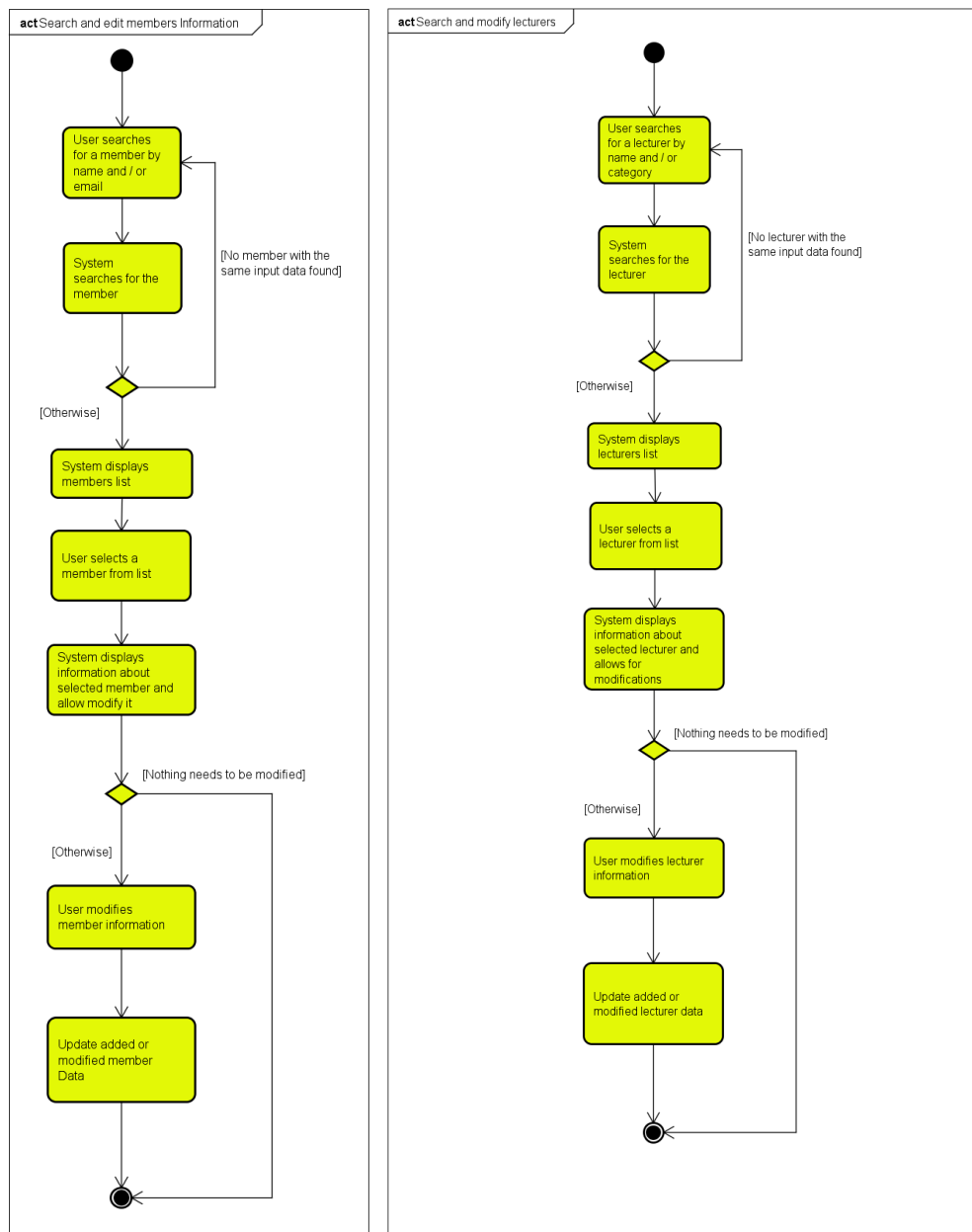
3.2 Description of Use Cases

The description of every Use Case is as follows:

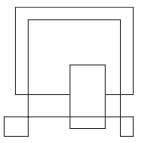
- **Create an event:** For creating a new event and storing it in the system.
- **Add lecturers and members:** For adding lecturers and members to the system.
- **Book an event:** Books an event for a member and stores the booking in the system.
- **Search and manage member information:** for searching and editing information about an existing member in the system.
- **Search and modify lecturers:** for searching and editing information about a specific lecturer in the system.
- **Search and manage events:** Search for events, and manage saved event attributes.



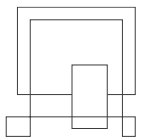
3.3 Activity Diagrams



The role of these diagrams is to show the detailed workflow of searching and modifying members and lecturers by the user. The user would be able to search by name for members or lecturers, and can search for lecturers by category and also search for members by email. The system searches for members or lecturers which match found data within that is already within the system. If the system doesn't find any lecturers or members with the same matching data, the user would have to search again, otherwise the system will display a list of matching results of either members or lecturers. The user will then be able to select a member or lecturer from the list in order to

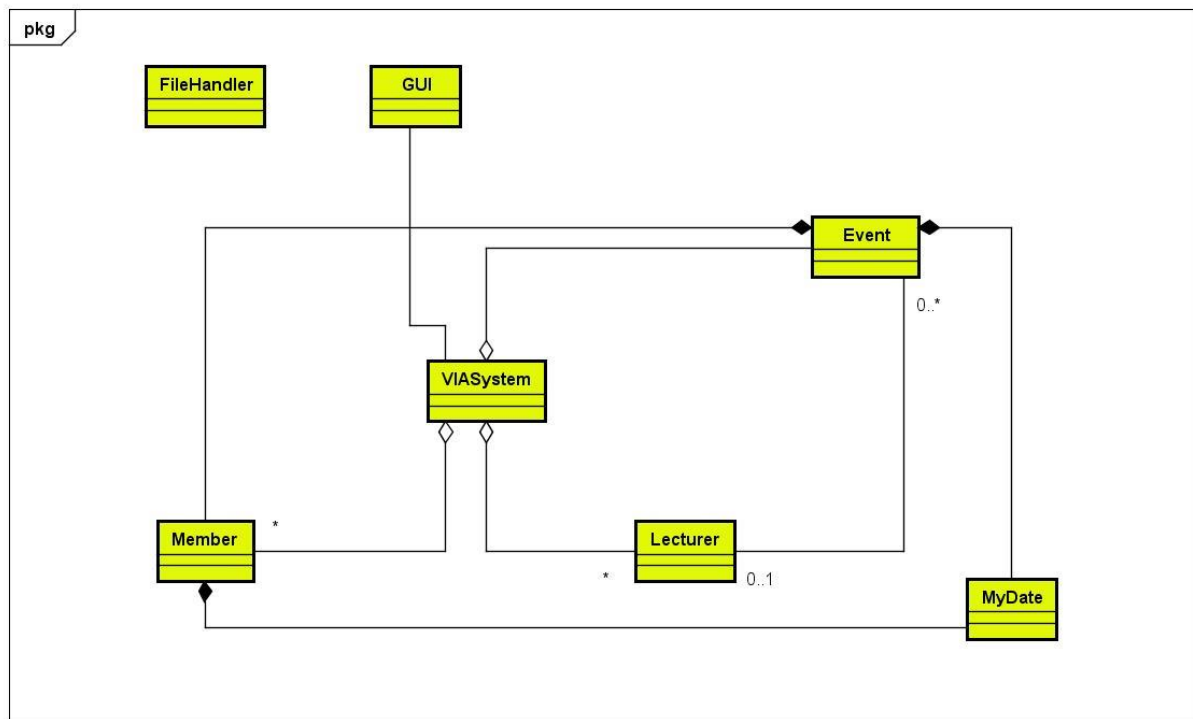


show and / or modify selected member or lecturer. If the user changed any information, the system will update all new information about that lecturer or member.



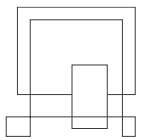
4 Design

4.1 Class Diagram

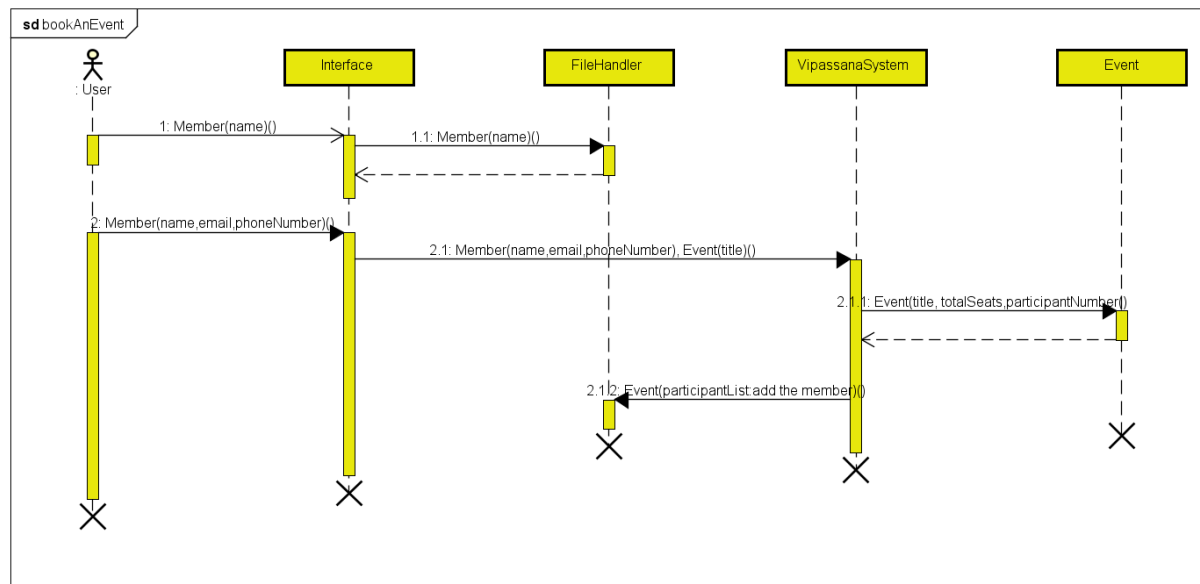


The simplified class diagram above is showing the relationships between the classes. The main part of the program is the **VIASystem** class which holds all the lists of lecturers, members, events, and all the methods necessary for creating, modifying, and deleting those objects. The event participants are stored in the event class as a list of members. The relationship between events and lecturers is association as all events need to have a lecturer except events of type “Journey”.

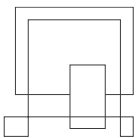
The **GUI** class represents the graphical user interface and the **FileHandler** class is there to handle files – create and save to them.



4.2 Sequence Diagram

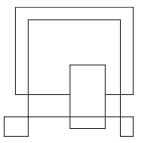


In this sequence diagram, the user books an event for a member. He needs to first search for the member by name. The name will then be searched within the system storage system (members file) and the interface will show a list of matching results (if any are found). The user can then select the member who wants to book the event. When he does, a message with that specific member and the wanted event (already chosen) being sent to the main class VipassanaSystem, which sends a message to the Event class in order to make sure that the event is finalized and not sold out (number of participants is less than total seats). Finally, the chosen member should be added to the event participants list and saved to the event file.

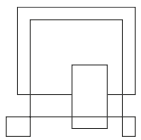


4.3 GUI

The GUI was developed using Window Builder. The main window is simple and consists of two tabs, the first is called “Lecturers and members information” and it allows adding and removing new members and lecturers, as well as searching for a specific one and editing or removing them.



The other tab is called “Event information”. It allows the user to easily create a new event (either finalized or not), find events to edit, search for lecturers to add to the event, and search for members to add to the participants list of the chosen event. However, the event must be finalized to add members to its participants list, and some fields are not mandatory to fill in when creating a new event if it is unfinalized.



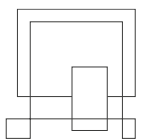
5 Implementation

The following code shows one of the most important functions in the system, which is adding a member to the participants list of an event:

```
{
    try
    {
        if (hollySystem.getEventByTitle(tempEventName)
            .getTotalSeats() <= listEventsParticipants.getItemCount()
            || hollySystem.getEventByTitle(tempEventName)
                .isFinalized() == false)
        {
            JOptionPane.showMessageDialog(new JFrame(),
                "Event is either sold out or unfinalized", "Error",
                JOptionPane.ERROR_MESSAGE);
        }
        else
        {
            if (!(listEventsMember.getSelectedItem() == null))
            {
                listEventsParticipants
                    .add(listEventsMember.getSelectedItem());
            }
            else
            {
                JOptionPane.showMessageDialog(new JFrame(),
                    "Select member to add for event!", "Error",
                    JOptionPane.ERROR_MESSAGE);
            }
        }
    }
    catch (NullPointerException ime)
    {
        JOptionPane.showMessageDialog(new JFrame(),
            "No member currently selected! Please select a member!",
            "Error", JOptionPane.ERROR_MESSAGE);
    }
}
```

A member needs to be searched and selected from the list before adding them to the list (NullPointerException is caught). If the selected event has total seats less or equal to the number of participants, or the selected event is not finalized, the system shows an error message, otherwise, the system adds the selected member to the participants list of the event.

Another important function in the system is Creating an event. The status must be defined when creating every new event (must be either finalized or unfinalized).



```
// When the status is not selected

if (!rdbtnFinalized.isSelected() && !rdbtnUnfinalized.isSelected())
{
    JOptionPane.showMessageDialog(new JFrame(),
        "You must select a status for the event!", "Error",
        JOptionPane.ERROR_MESSAGE);
}
```

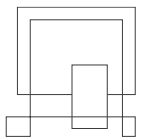
When the event is finalized, the system checks if all required fields are filled in correctly as shown below, and then starts creating the new event. However, not all fields are mandatory to fill in when creating an unfinalized event.

```
// When the event is finalized
if (rdbtnFinalized.isSelected())
{
    // Check if all fields have variables.
    if (!(eventTitle.getText().isEmpty()
        || eventCategory.getText().isEmpty()
        || eventType.getText().isEmpty()
        || eventTotalSeats.getText().isEmpty()
        || eventPrice.getText().isEmpty()
        || !eventTotalSeats.getText().matches("[0-9]+")
        || !eventPrice.getText().matches("[0-9]+")
        || !eventEndYear.getText().matches("[0-9]+")
        || !eventStartYear.getText().matches("[0-9]+")))
    {
        MyDate dateStart = new MyDate(0, 0,
            Integer.parseInt(eventStartDay.getValue().toString()),
            Integer.parseInt(eventStartMonth.getValue().toString()),
            Integer.parseInt(eventStartYear.getText()));

        MyDate dateEnd = new MyDate(0, 0,
            Integer.parseInt(eventEndDay.getValue().toString()),
            Integer.parseInt(eventEndMonth.getValue().toString()),
            Integer.parseInt(eventEndYear.getText()));

        Event event = new Event(eventCategory.getText().toString(),
            eventType.getText(), eventTitle.getText(),
            textDescription.getText(), eventLecturers, dateStart,
            dateEnd, true,
            Integer.parseInt(eventTotalSeats.getText()),
            Integer.parseInt(eventPrice.getText()),
            textFeedback.getText(), eventMembers);
        hollySystem.addEvent(event);
        JOptionPane.showMessageDialog(new JFrame(),
            "Event has been added to database!", "Done!",
            JOptionPane.OK_CANCEL_OPTION);
    }
}
```

```
Event event = new Event(eventCategory.getText().toString(),
    eventType.getText(), eventTitle.getText(),
    textDescription.getText(), eventLecturers, dateStart,
    dateEnd, true,
    Integer.parseInt(eventTotalSeats.getText()),
    Integer.parseInt(eventPrice.getText()),
    textFeedback.getText(), eventMembers);
hollySystem.addEvent(event);
JOptionPane.showMessageDialog(new JFrame(),
    "Event has been added to database!", "Done!",
    JOptionPane.OK_CANCEL_OPTION);
```



6 Test

6.1 Core functionality:

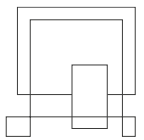
#	Functionality description	Status
1	GUI window displays upon runtime of application	Working
2	System creates new storage files if previous ones are not found	Working
3	Data persistence is consistent on program runtime	Working
4	Window size and window elements are resizable in full screen	Not implemented
5	Usage of separate JFrame windows for given sections of system	Not implemented
6	Invalid data input safeguards	Working
7	Program works with full date information	Partly-implemented

6.2 “Lecturer and Member information” tab:

#	Functionality description	Status
1	Reading member and lecturer information from storage system	Working
2	Writing member and lecturer information to storage system	Working
3	Adding a member / lecturer	Working
4	Finding a member / lecturer	Working
5	View of all members / lecturers	Working
6	Removing a member / lecturer	Working
7	Modifying current member / lecturer information	Working
8	Displaying member / lecturer list	Working

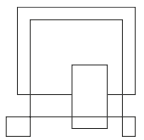
6.3 “Event information” tab:

#	Functionality description	Status
1	Reading event information from storage system	Working
2	Writing event information to storage system	Working
3	Adding an event	Working
4	View all events	Working
5	View finalized events	Working
6	View unfinalized events	Working
7	Search for matching member / lecturer within storage system	Working
8	Association of member / lecturer to event	Working
9	Updating / removing an event	Working
10	Member event preferences	Not implemented
11	Member emailing system	Not implemented



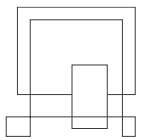
➤ **Conclusion:**

Critical functionality has been implemented and program functionality is normal. File persistence is utilized by the system on every runtime – saving and reading data, data differentiation, information reading and input collection.



7. Results and Discussion

The system is implemented properly with all necessary features to handle files and do the job covering for all important requirements.



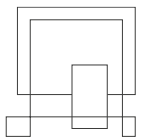
8 Conclusion

The system covers all necessary functionality and is in compliance with requested requirements by which it is almost completely functional to use by the end user.

The program makes use of event date setting, however, it cannot make use of whole time saving as this feature is partly implemented. In essence, the user is unable to specify a given hour and minute for any event that they add or modify.

#	Program functionality description	Status
1	The user should be able to create, modify and remove events within the system.	Fulfilled
2	The system should be able to store the customer's name, phone number, payment year, date of membership and e-mail address.	Fulfilled
3	The system should be able to list all events.	Fulfilled
4	The system should provide the user the possibility to leave feedback on events on behalf of customers.	Fulfilled
5	The user should be able to book events within the system.	Fulfilled
6	The system should have a member list that includes name, phone number, date of membership and year of payment, and optionally an e-mail address.	Fulfilled
7	The user should be able to search for lecturers and events within the system.	Fulfilled
8	The user should be able to search in a specific category within the system.	Fulfilled
9	The user should be able to modify member and lecturer information within the system.	Fulfilled
10	The user should be able to list all members within the system.	Fulfilled
11	The user should be able to add new members and lecturers within the system.	Fulfilled
12	The user should be able to finalize events within the system.	Fulfilled
13	The user should be able to leave feedback on a selected event.	Fulfilled

All requirements that were defined have been fulfilled within this project.



9 Project future

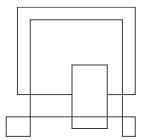
Changes to the project in the future can include code optimization – better structuring, avoidance of code clutter, extension of java classes and division of parent classes.

New features:

The system implementation would benefit from a better GUI and as such, making use of the JavaFX library would be the best for how GUI design would be better. This, in combination with tables, is a much better choice instead of relying on Swing for handling the interface.

Multiple selection of members and lecturers for adding to an event can also be added so that the user can choose from more than one member at a time.

An email messaging system would also be a positive addition to the program by which the user can select members and send information about the event to them.



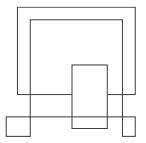
10 References

Roberta A. Szekeres, Eleanor H. Wertheim. 2014. Evaluation of Vipassana Meditation Course Effects on Subjective Stress, Well-being, Self-kindness and Mindfulness in a Community Sample: Post-course and 6-month Outcomes. [online]

Available at: <http://onlinelibrary.wiley.com.ez-aaa.statsbiblioteket.dk/2048/doi/10.1002/smi.2562/full>

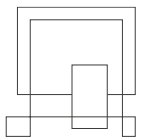
Front Psychol. 2016. Psychological Effects of a 1-Month Meditation Retreat on Experienced Meditators: The Role of Non-attachment. [online]

Available at: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5149565/>



11 List of Appendixes

- Project Description
- User Guide
- Design class diagram



Original Project Description

Vipassanā – Insight Awareness (VIA)

Students

Angel Petrov - 266489

Rytis Kunigiskis - 266109

Diyar Hussein - 266352

Rodrigo Pereira - 269084

Supervisors

Mona Wendel Andersen

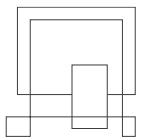
Michael Vuff

ICT Engineering

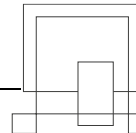
1st Semester

02/10/2017

Table of content



1	Background description	24
2	Definition of purpose	25
3	Problem Statement	26
4	Delimitation	27
5	Choice of models and methods	28
6	Time schedule	29
7	Risk assessment	30
8	Sources of Information	31
9	Appendices (including Group Contract)	31

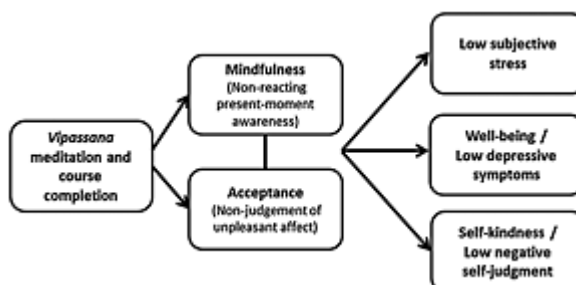


1 Background description

Vipassanā - Insight Awareness (VIA) is a centre for spiritual events originally with a base in the Buddhist principles of meditation as an insight with awareness of what is happening as it happens. They organize spiritual events, send out newsletters and sometimes articles to their members.

Vipassanā does not store any trace of lecturers, members, plans for events, work done by its council or whom they promised advertisements free of charge. This gives the rise of potential loss of clients which in turn opens a liability in the way that the organisation operates, its potential achievements and overall stability. Studies have proven that Vipassanā's meditation is rewarding to the individual - benefits such as improvement of psychological health caused the means to solve the problem of current situation of Vipassanā, the project will benefit the wide audience of people. Furthermore, there are sources which claim that Vipassanā's practices yield a more prosperous way of living as a conclusion from studies on a wide range of audiences who have been exposed to meditation courses with a wide variety of time spans. It can be concluded that stress management is an important area of concern and it should be taken care off because it is not only beneficial for the organization but at the same time it also improves individual life of employees. The following diagram shows the outcomes on people which have been subjected to a meditation course lasting six months.

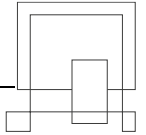
Evaluation of *Vipassanā* Meditation Course Effects on Subjective Stress, Well-being, Self-kindness and Mindfulness in a



Community Sample: Post-course and 6-month Outcomes

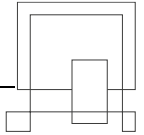
“Sometimes a single Vipassana course can give someone enough direct experience of the truth inside that for them the path is clear: “This is where I want to go. This is what I want to be.” For others it takes longer.” - Independent Publishers Group I.H 2003 9781928706212

The quote above represents the thoughts of a person who has underwent the Vipassanā course clearly pointing out that a person's perspective changes.



2 Definition of purpose

The definition of purpose for this project is to construct a system that has the ability to keep track of records which will allow Vipassana to improve its data management.



3 Problem Statement

Overall problem:

The current and future problems that the group will encounter along the way will be the lack of information that the system will process, the necessary automation methods - if, for example, a specific part of people need to be selected or contacted - what would be a necessary pre-requirement for the system to go into effect.

The questions regarding the problem statement are the following:

Q: How will the backend of the system tie with the GUI (Graphical User Interface) and what possibilities will the system have for the GUI to offer?

The possibilities of the system and the GUI structure need to be appropriate with one another for there to be a consistent logic in the systems' possibilities. Some of the possibilities are still unknown and some may likely change as the project moves forward.

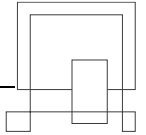
Q: How will the system know who is going to have higher privileges than the other? The system will need to differentiate between a user and a company representative so that both have easy access

Q: How can the system store the different events and members with all of their personalized information?

Based on the actual collected information, the system needs to meet three different criterias - information handling, events handling and categorization of the said personalized user information.

Q: How can the user search for lecturers in a given category?

A good and easy-to-use user interface is a very important part in every application and system. Therefore the group should spend some time to insure an easy user experience with a good search engine.

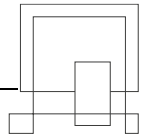


4 Delimitation

The system and the project are aimed towards the software market and are in turn developed for the clients who have requested such a service to be created - Vipassanā - Insight Awareness.

What the assigned group will firmly focus on are the technical problems, For example, how to make the system better. Otherwise, there will not be focus on problems such as searching for the most wished lectures and events.

Another thing that the group will not focus on is a good looking user interface, as the group will focus on adding key functionality to the systems' abilities.



5 Choice of models and methods

Internal group event handling:

The project's strategy of overcoming potential problems will have a direct impact on its outcomes. The aim is to gain a solid knowledge base for the team that is working on the assigned project and focus that gained knowledge whilst researching problems which will be encountered during its development. The focus in this project is going to be on creating all assets, testing and finalizing the working system - in other words words: **events handling**. This is a critical aspect of project management as well as workflow management. Keeping track of progress of the workflow is vital as it yields a better perception of the overall outcome. The methodology behind this will be to have group work as well as individual handling of work and / or upcoming project events. The responsibility of the events handling, time planning, and workload distribution will be assigned to Angel Petrov which will in turn plan ahead of time and see through upcoming tasks and solve any obstacles related to the group work distribution.

System readability and management:

Another partial problem is making the system as **easy to understand** as possible so that the end users don't have any problems such as encountering bugs or compatibility issues. Managing code is also within this category - adding comments for "what goes where" or "what this thing does" is a good way of understanding things that fellow colleagues have implemented, however not first priority. A better outcome is expected as others are going to have a better understanding of what the person who implemented that specific area of the project actually meant and if they see the need, they will have a better idea of how to improve that specific area. The main person who will be responsible for this area will be Rytis Kunigiskis, however everyone will assume this role to a certain extent but the responsible person for this area will have a focus on going through all the code. This said person will have the opportunity to go through all of the code and discuss what is right or wrong and propose to add or remove specific parts from the structure of the system. A rough estimated workload for this part will be around 10 percent or 55 hours of the overall workload of 550 hours but overall the timing will solely rely on which parts of the project are currently being evaluated, their length and meaning within the project (some parts do not need to be explained, others do).

Finding sponsors for newsletters:

The system will not only store sponsors that have been sponsoring Vipassana successfully before, but also will allow to maintain the list, add new sponsors or delete the old ones and add or edit the description of each sponsor. It will allow the customer to keep track of their sponsors and quickly get information about them, give them feedback about the sponsors for the categories that they show interest in. Diyar Hussein will take responsibility for this point and estimated workload for this point should be around 5 hours or around 5 percent of the overall workload.

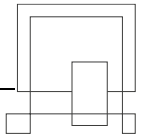
Finding non-finalized events to finalize them:

Planning events requires a lot of information and quick access to it with a good search engine will provide fast planning to finalize all non-finalized events. The user will be able to review the status of any event and see all required details about it. Rodrigo Pereira will hold responsibility for this point. Estimated time for fulfilling this task should be around 83 hours or around 15 percent.

Storing needed data for the system:

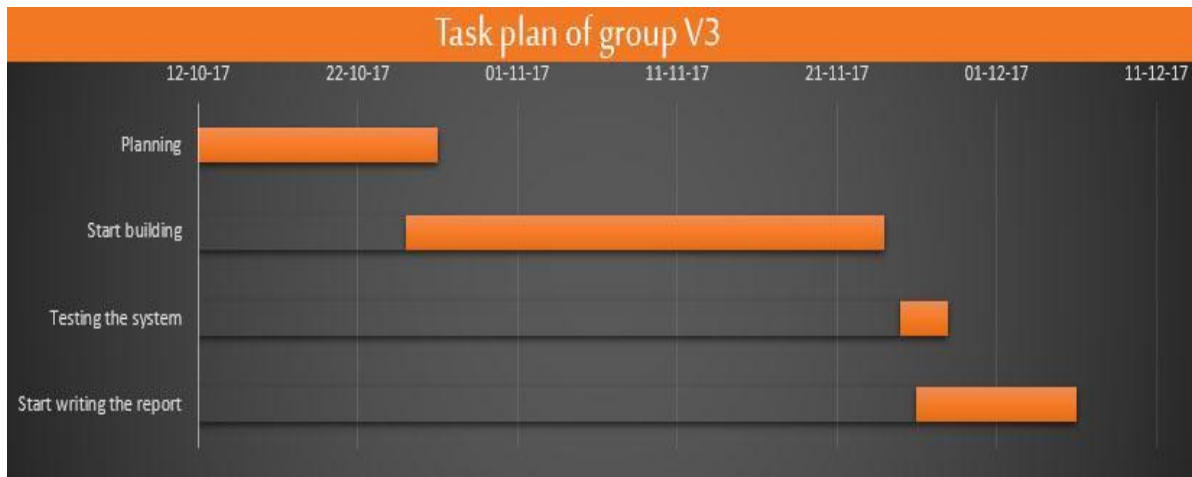
Data will be stored locally. This will ensure data protection. Everyone in the group is responsible for this point. Estimated work hours around 25 hours or 5 percent.

		Version: 1.00	
File: Group_3_Project_Report.docx		Last modified: 2017-12-19	28 of 58

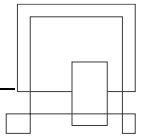


6 Time schedule

Our group will work on the project based on the [Waterfall](#) model which will in turn yield a progressive style of software development throughout the different project stages as opposed to using models such as, for example, [SCRUM](#) or [Kanban](#).

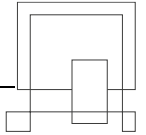


The following table represents the assigned group's workload and work span in a given category based on the Waterfall model:



7 Risk assessment

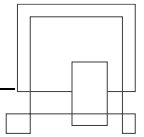
Risks	Description	Likelihood	Severity	Risk mitigation	Identifiers	Responsible
Running out of time	Time management is a necessity	3	4	Work in the specified work time and plan	Coming up project deadline, stalling	Group 3
No knowledge of how to perform some project parts	To get the best out of the system, knowledge has to be well collected	2	3	Research and ask questions	No initial planning	Group 3
Project team instability	Team stability is a key factor to success and vice versa	4	5	Insure individual member well-being	Member absence, uncertainty, unreliability	Group 3
Infringement on third party sources	Relying on proprietary content for "inspiration"	1	3	Avoid at all costs	External code used that is guarded by a proprietary license	Group 3



8 Sources of Information

Background description:

- ❖ [Source № 1](#) (login required)
- ❖ [Source № 2](#)
- ❖ [Source № 3](#)
- ❖ [Source № 4](#) (login required) - [alternative link](#)



VIPASSANA MANAGEMENT SYSTEM - VIA

User Guide

Supervisors:

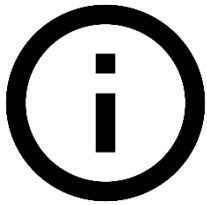
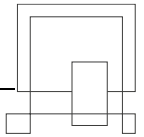
Mona Wendel Andersen

Michael Viuff

19 / 12 / 2017

Group 3

Student Name	Student No
Angel Petrov	266489
Rytis Kunigiskis	266109
Rodrigo Pereira	269084
Diyar Hussein	266352



Preface:

The following user guide gives a step by step explanation to correctly utilize all features of the system.



WARNING:

Before you start using this software, please, read the following information carefully:

Runtime requirements:

This program makes use of the Java Runtime Environment by software developer Oracle which is necessary in order for the program to operate correctly. You can find this software at oracle.com/downloads.

Program functionality:

This program makes use of an additional data storage system which reads and writes additional files for the program to function properly.



Trademark information:

The Java Runtime Environment and Java are trademarks of Sun Microsystems, Inc. in the U.S. and certain other countries.

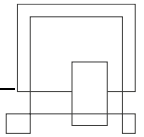
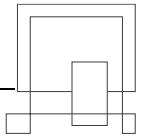


Table of contents:

Overview of program.....	4
Lecturer & Member information.....	5
Members.....	6
Lecturer data.....	10
Lecturers.....	11
Event Information.....	15
Sections of Event information.....	16
Create and update events.....	17
Events.....	18
Add lecturers and participants.....	20



1. Overview of the program:

The screenshot shows the 'VIA Management System' window with the 'Lecturer & Member information' tab selected. The interface is split into two main columns for 'Lecturers' and 'Members'.

Lecturers Section:

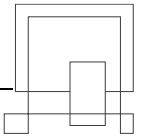
- Search:** Name: Find; Category: View lecturer list
- List:** Lecturer list (empty box)
- Data Entry:** Name: Remove lecturer; Email: Add lecturer; Category: Update information; Free advertisement: ☐ Yes ☐ No

Members Section:

- Search:** Name: Find; Email: View all members
- List:** Member list (empty box)
- Data Entry:** Name: Remove member; Email: Add member; Phone Number: ; Payment Year: ; Date of membership: Day: (1), Month: (1), Year: ; Update information

Description:

This is the start screen for the program. The main tab of the program is entitled “**Lecturer & Member information**” and contains four sections: “**Member data**”, “**Members**”, “**Lecturer data**”, and “**Lecturers**”.



2. Lecturer & Member information:

VIA Management System

Lecturer & Member information | Event information

Lecturers

Name:

Category:

Lecturer list

Members

Name:

Email:

Member list

Lecturer data

Name:

Email:

Category:

Free advertisement
☐ Yes ☐ No

Member data

Name: Email:

Phone Number:

Payment Year:

Date of membership
Day: Month: Year:

1

3. Member data:

Within the member section, you are able to add new members, modify existing information and remove a member.

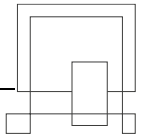
Member data

Name: Email:

Phone Number:

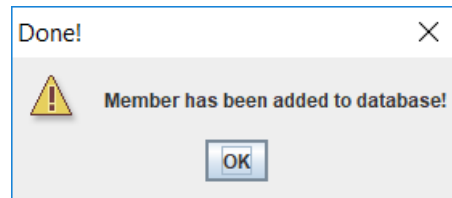
Payment Year:

Date of membership
Day: Month: Year:

**To add a member:**

- Fill the name field with the member's information;
- Fill the email field with the member's information;
- Fill the phone number field with the information provided by the customer;
- Write the payment year in the payment year field;
- Select the date of membership for the member which includes day, month and year;
- Press the "Add member" button to store the member's information.

Upon successful input of information, you will be presented with the following message:



Information about the member has now been stored and you may close this window.

2

4. Members:

To find, remove or update an existing member, you will need to have added that member first before being able to modify member information. You are able to find a member by either name or email.

Members

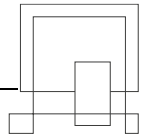
Name:

Email:

Find

View all members

Member list

**To find a member:**

- Enter relevant search information about the member in the name field (Optional if step b has been done);
- Enter email address of member in the email field (Optional if step a has been done);
- Click on the “Find” button;

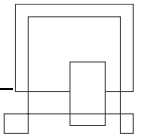
Upon successful search, you will be presented with the desired member in the member list:

The screenshot shows a window titled "Members". It has two input fields: "Name:" with the text "Angel Petrov" and "Email:" with the text "angel@aa.com". To the right of the "Name:" field is a "Find" button, and to the right of the "Email:" field is a "View all members" button. Below these fields is a section titled "Member list" which contains a text box displaying the search results: "Name: Angel Petrov, Phone Number: 71902066, Email: angel@aa.com, Payment". At the bottom of the text box is a horizontal scrollbar.

Viewing all members:

- Click on the “View all members” button;
- The program will show all available members that have been added to the program, prior to viewing all members:

The screenshot shows the same "Members" window. The "Name:" and "Email:" fields are empty. The "Find" button is still present. The "View all members" button is highlighted. The "Member list" section now displays a list of four members: "Name: Angel Petrov, Phone Number: 71902066, Email: angel@aa.com, Payment", "Name: Rytis Kunigiskis, Phone Number: 71902432, Email: rytisK@bb.com, Paym", "Name: Rodrigo Pereira, Phone Number: 71905569, Email: rodri@cc.com, Paymei", and "Name: Diyar Hussein, Phone Number: 71905811, Email: diyar@dd.com, Paymen". A horizontal scrollbar is visible at the bottom of the list.



Removing a member from the member list:

- Select a member that you wish to remove from the member list;
- In the “Members data” section, click on the “Remove member” button:

Members

Name: Find

Email: View all members

Member list

Name: Angel Petrov, Phone Number: 71902066, Email: angel@aa.com, Payment
Name: Rytis Kunigiskis, Phone Number: 71902432, Email: rytisK@bb.com, Paym
Name: Rodrigo Pereira, Phone Number: 71905569, Email: rodri@cc.com, Payme
Name: Diyar Hussein, Phone Number: 71905811, Email: diyar@dd.com, Paymen

< >

Member data

Name: Email:

Phone Number:

Payment Year:

Date of membership

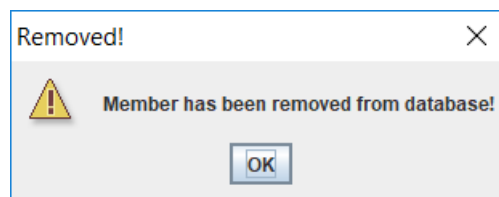
Day: Month: Year:

Remove member

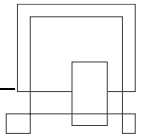
Add member

Update information

Upon successful deletion of a member, you will be presented with the following message:



Information about the member has now been updated and you may close this window.



Updating information about member:

- Select a member that you wish to update information about from the member list and edit any desired information about the member;
- Click on the “Update information” button to save updated information.

Members

Name: Find

Email: View all members

Member list

Name: Angel Petrov, Phone Number: 71902066, Email: angel@aa.com, Payment Year: 2017
Name: Rytis Kunigiskis, Phone Number: 71902432, Email: rytisK@bb.com, Payment Year: 2017
Name: Rodrigo Pereira, Phone Number: 71905569, Email: rodri@cc.com, Payment Year: 2017
Name: Diyar Hussein, Phone Number: 71905811, Email: diyar@dd.com, Payment Year: 2017

Member data

Name: Email:

Phone Number:

Payment Year:

Date of membership

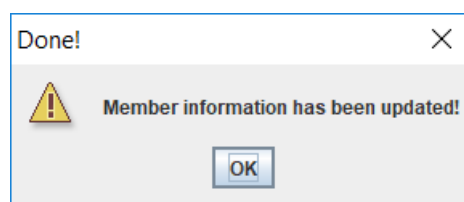
Day: Month: Year:

Remove member

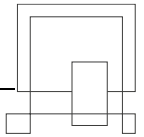
Add member

Update information

Upon successful update of a member information, you will be presented with the following message:



Information about the member has now been updated and you may close this window.



3

5. Lecturer data:

Within the lecturers section, you are able to add new lecturers, modify existing information and remove a lecturer.

Lecturer data

Name:

Email:

Category:

Free advertisement
☐ Yes ☐ No

Remove lecturer

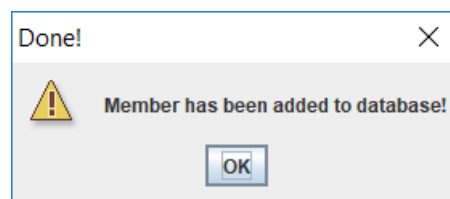
Add lecturer

Update information

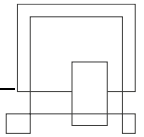
To add a lecturer:

- Fill the name field with the lecturer's information;
- Fill the email field with the lecturer's information;
- Write the category that the lecturer will be in;
- Write the payment year in the payment year field;
- Select the date of membership for the member which includes day, month and year;
- Press the "Add member" button to store the member's information.

Upon successful input of information, you will be presented with the following message:



Information about the member has now been stored and you may close this window.



4

6. Lecturers:

To find, remove or update an existing lecturer, you will need to have added that lecturer first before being able to modify lecturer information. You are able to find a lecturer by either name or category.

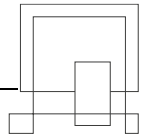
The screenshot shows a web form titled "Lecturers". It has two input fields: "Name:" and "Category:". To the right of the "Name:" field is a "Find" button. To the right of the "Category:" field is a "View lecturer list" button. Below these fields is a section titled "Lecturer list" which contains a large empty rectangular box for displaying results.

To find a lecturer:

- Enter relevant search information about the lecturer in the name field (Optional if step b has been done);
- Enter email address of member in the email field (Optional if step a has been done);
- Click on the "Find" button;

Upon successful search, you will be presented with the desired lecturer in the lecturer list:

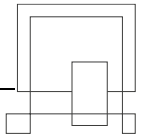
This screenshot shows the same "Lecturers" form as before, but with search results. The "Name:" field contains "Rytis Kunigiskis" and the "Category:" field contains "Astrology". The "Find" and "View lecturer list" buttons are still present. The "Lecturer list" section now displays a single entry: "Name: Rytis Kunigiskis, Email: rytisk@bb.com, Category: Astrology, Advertis". Below the text is a horizontal scrollbar.

**Viewing all lecturers:**

- Click on the “View lecturer list” button;
- The program will show all available lecturers that have been added to the program, prior to viewing all lectures:

Lecturers
Name:
Category:
Lecturer list
Name: Angel Petrov, Email: angel@aa.com, Category: Astrology, Advertisem
Name: Rytis Kunigiskis, Email: rytisK@bb.com, Category: Reincarnation, Ad
Name: Rodrigo Pereira, Email: rodi@cc.com, Category: Karma, Advertiseme
Name: Diyar Hussein, Email: diyar@dd.com, Category: AltHealth care, Adver

< >

**Removing a lecturer from the member list:**

- Select a lecturer that you wish to remove from the lecture list;
- In the “Lecturer data” section, click on the “Remove lecturer” button:

Lecturers

Name: Find

Category: View lecturer list

Lecturer list

Name: Angel Petrov, Email: angel@aa.com, Category: Astrology, Advertisem
Name: Rytis Kuniqiskis, Email: rytisK@bb.com, Category: Reincarnation, Ad
Name: Rodrigo Pereira, Email: rodi@cc.com, Category: Karma, Advertisem
Name: Diyar Hussein, Email: diyar@dd.com, Category: AltHealth care, Adver

< >

Lecturer data

Name:

Email:

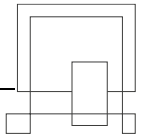
Category:

Free advertisement
☒ Yes ☐ No

Remove lecturer

Add lecturer

Update information



Updating information about lecturer:

- Select a lecturer that you wish to update information about from the lecturer list and edit any desired information about the lecturer;
- Click on the “Update information” button to save updated information.

Lecturers

Name: Find

Category: View lecturer list

Lecturer list

Name: Angel Petrov, Email: angel@aa.com, Category: Astrology, Advertisement
Name: Rytis Kunigskis, Email: rytisk@bb.com, Category: Reincarnation, Advertisement
Name: Rodrigo Pereira, Email: rodi@cc.com, Category: Karma, Advertisement
Name: Diyar Hussein, Email: diyar@dd.com, Category: AltHealth care, Advertisement

< >

Lecturer data

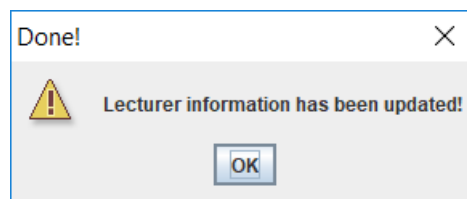
Name: Remove lecturer

Email: Add lecturer

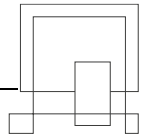
Category: Update information

Free advertisement
☒ Yes ☐ No

Upon successful update of a member information, you will be presented with the following message:



Information about the lecturer has now been updated and you may close this window.



7. Event Information:

VIA Management System

Lecturer & Member information | **Event information**

Events

Title: **Find** **View all events**

Type: **View finalized events** **View unfinalized eve...**

Remove selected event

Add lecturers and participants

Lecturer's name: **Search** Member's name: **Search**

Lecturer list

Member list

Add selected lecturer to the event **Book this event for the selected member**

Feedback:

Create and update events

Title: Start date Day: Month: Year:

Category: End date Day: Month: Year:

Type: Description:

Total Seats: Price:

Status: ☐ Finalized ☐ Unfinalized

Add event **Update information**

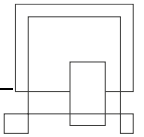
Lecturers for event

Participating members: **Remove selected lecturer**

Remove selected member

Description:

The illustration above is an overview of the event tab which contains three sections: “**Create and update events**”, “**Events**” and “**Add lecturers and participants**”.



8. Sections of Event information:

VIA Management System

Lecturer & Member information **Event information**

Events

Title: **Find** **View all events**

Type: **View finalized events** **View unfinalized eve...**

Remove selected event

Add lecturers and participants

Lecturer's name: **Search** Member's name: **Search**

Lecturer list **Add selected lecturer to the event** Member list **Book this event for the selected member**

Create and update events

Title: Start date Day: Month: Year: End date Day: Month: Year:

Category: Type: Description:

Total Seats: Price: Status: ☐ Finalized ☐ Unfinalized

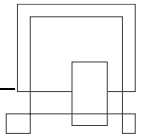
Add event **Update information**

Lecturers for event

Participating members: **Remove selected lecturer**

Remove selected member

Feedback:



1

9. Create and update events:

In the create and update events section, you are able add information for an event, its starting and ending time, a description for the event and leave feedback.

Creating an event:

Create and update events

Title: Start date Day: Month: Year:

Category: End date Day: Month: Year:

Type: Description:

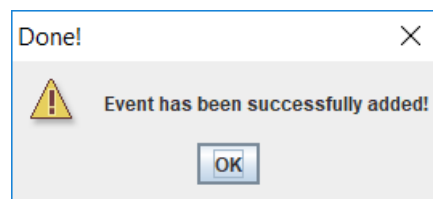
Total Seats: Price:

Status: ☐ Finalized ☐ Unfinalized

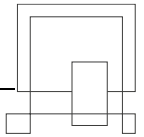
Feedback:

- Fill the title field with the title of the event;
- Write in which category the event is within the category field;
- Specify the type of event in the type field;
- Fill the total seats for the event in the total seats field;
- Specify the price for the event in the price field;
- Choose one of the available buttons - "Finalized" or "Unfocalized" depending on event status;
- Select the start day, month and fill in the year field with the event start date;
- Select the end day, month and fill in the year field with the event end date;
- Optionally, give a description for the given event in the "Description" field;
- Optionally, leave feedback for the given event in the "Feedback" field;
- Press the "Add event" button to save event information;

Upon successful input of information, you will be presented with the following message:



Information about the lecturer has now been updated and you may close this window.



2

10. Events

In the events section, you are able to find events by either title, type or both.

Finding an event:

Events

Title: Find View all events

Type: View finalized events View unfinalized eve...

- Enter the title of the event that you would like to find in the title field (Optional if step b has been done);
- Enter the type of the event that you would like to find in the type field (Optional if step a has been done);
- Click on the “Find” button to find events based on the search preferences that you provided.

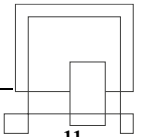
Upon successful search, you will be presented with the desired lecturer in the lecturer list:

Events

Title: Vipassana Find View all events

Type: Meditation View finalized events View unfinalized eve...

Title: Vipassana, Category: Meditation, Type Opening, Descri



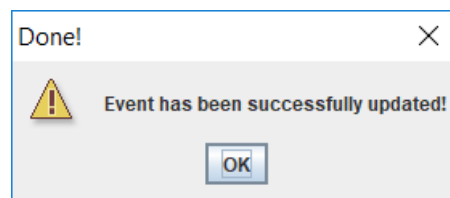
- d. To get a list of all events that have been added, click on the button “View all events”;
- e. To get a list of events that have been finalized, click on the button “View finalized events”;
- f. To get a list of events that have not been finalized yet, click on the button “View unfinalized events”.

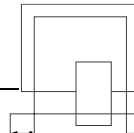
Editing event information:

- a. Select an event that you wish to update information about from the events section;
- b. Click on the “Update information” button to save updated information.

The screenshot displays the VIA Management System interface. The top section, titled 'Events', includes search filters for Title (Vipassana) and Type (Meditation), along with buttons for 'Find', 'View all events', 'View finalized events', and 'View unfinalized events'. Below these is a table listing events, with the first row highlighted: 'Title: Vipassana, Category: Meditation, Type: Opening, Description: Welcome to Vipassana Insight Awareness!'. An orange circle labeled 'a' points to this row. The bottom section, titled 'Create and update events', contains fields for Title (Vipassana), Start date (Day: 26, Month: 2, Year: 2018), End date (Day: 27, Month: 2, Year: 2018), Category (Meditation), Type (Opening), Total Seats (256), and Price (50). It also has radio buttons for Status (Finalized, Unfinalized) and buttons for 'Add event' and 'Update information'. A blue circle labeled 'b' points to the 'Update information' button. A feedback box on the right shows a message: 'Waiting for the event with high expectations! - Angel Petrov - 17/12/2017;'.

Upon successful input of information, you will be presented with the following message:





Information about the lecturer has now been updated and you may close this window.

3

11. Add lecturers and participants:

The lecturers and participants section provides the possibility to add lectures and participants for a selected event that have already been added under the “Lecturer & Member information” in the sections “Lecturer data” and “Member data”.

Searching for lecturers:

- Fill in the lecturer's name in the designated field;
- Press the “Search” button to search with the given criteria;

Searching for members:

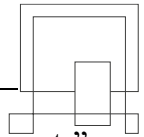
- Fill in the member's name in the designated field;
- Press the “Search” button to search with the given criteria;

Assigning a lecturer to an event:

- From the lecturer list, select a desired lecturer;
- Click on the “Add selected lecturer to the event” button;



Note: Lecturer will now be added in the “Lecturers for event” list;

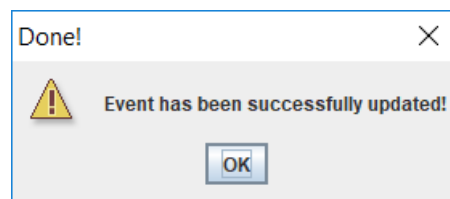


- c. Click on the “Update information” button in the “Create and update events” section;

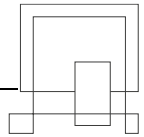
The screenshot displays two main sections of the application interface:

- Add lecturers and participants (Yellow box):**
 - Lecturer's name:** Search field with a "Search" button.
 - Member's name:** Search field with a "Search" button.
 - Lecturer list:** A list showing "Name: Angel Petrov, Email: angel@aa.com, C". Below the list is a button "Add selected lecturer to the event".
 - Member list:** A list showing "Name: Rytis Kunigiskis, Phone Number: 719024". Below the list is a button "Book this event for the selected member".
 - Feedback:** A text area containing "Waiting for the event with high expectations! - Angel Petrov - 17/12/2017;".
- Create and update events (Blue box):**
 - Title:** Vipassana
 - Category:** Meditation
 - Type:** Opening
 - Total Seats:** 256
 - Price:** 50
 - Status:** ☐ Finalized ☒ Unfinalized
 - Start date:** Day: 26, Month: 2, Year: 2018
 - End date:** Day: 27, Month: 2, Year: 2018
 - Description:** Welcome to Vipassana Insight Awareness! Here, you are able to register and follow the latest news in regards to the grand opening of the Vipassana meditation center! As a token of our appreciation, we invite you to...
 - Buttons:** "Add event" and "Update information" (highlighted by a blue circle labeled 'c').

Upon successful input of information, you will be presented with the following message:



Information about the lecturer has now been updated and you may close this window.



Assigning a member to an event:

- From the member list, select a desired member;
- Click on the “Book this event for the selected member” button;



Note: Member will now be added in the “Lecturers for event” list;

- Click on the “Update information” button in the “Create and update events” section;

Add lecturers and participants

Lecturer's name: Member's name:

Lecturer list:

Member list:

Name: Rytis Kunigskis, Phone Number: 719024

Feedback: Waiting for the event with high expectations! - Angel Petrov - 17/12/2017;

Create and update events

Title: Start date Day: Month: Year:

Category: End date Day: Month: Year:

Type: Description:

Welcome to Vipassana Insight Awareness! Here, you are able to register and follow the latest news in regards to the grand opening of the Vipassana meditation center! As a token of our appreciation, we invite you to...

Total Seats: Price:

Status: ☐ Finalized ☒ Unfinalized

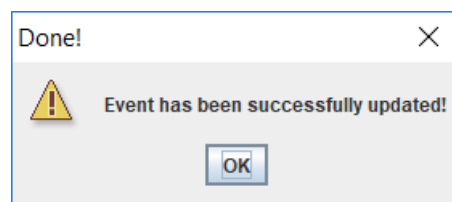
Lecturers for event

Name: Angel Petrov, Email: angel@aa.com, Category: As :

Participating members:

Name: Angel Petrov, Phone Number: 71902066, Email: ange

Upon successful input of information, you will be presented with the following message:

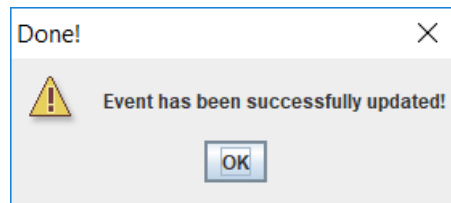
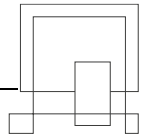


Information about the event has now been updated and you may close this window.

Removing a lecturer from an event:

- From the “Lecturers for event”, select the desired lecturer that you would like to remove for the currently selected event;
- Click on the “Remove selected lecturer”;
- Click on the “Update information” button in the “Create and update events” section;

Upon successful input of information, you will be presented with the following message:

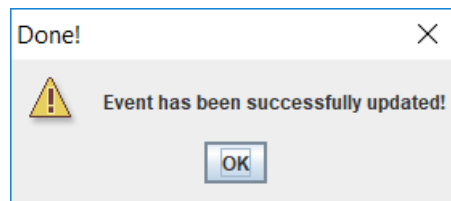


Information about the event has now been updated and you may close this window.

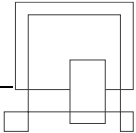
Removing a member from an event:

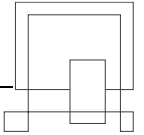
- a. From the “Lecturers for event”, select the desired lecturer that you would like to remove for the currently selected event;
- b. Click on the “Remove selected lecturer”;
- c. Click on the “Update information” button in the “Create and update events” section;

Upon successful input of information, you will be presented with the following message:



Information about the event has now been updated and you may close this window.





Thank you for consulting with this manual! We hope that you enjoy using this software!

Version 1.06 – 18 / 12 / 2017

		Version: 1.00	
File: Group_3_Project_Report.docx	Last modified: 2017-12-19	56 of 58	

