MeetMe Planner
Requirements Definition

Version 2.0

Version	Description	Date	Authors
0.1	Initial Draft	04/11/2015	Mohamed Mehdi Kaabi
0.2	Update	05/11/2015	Mohamed Mehdi Kaabi Milica Jovanovic
0.3	Update	06/11/2015	Carolina Casali Armindo Carvalho Mohamed Mehdi Kaabi
0.4	Update	10/11/2015	Mohamed Mehdi Kaabi
1.0	Update	18/11/2015	Mohamed Mehdi Kaabi Danijel Sokač
2.0	Update	14/01/2015	Mohamed Mehdi Kaabi

1. Description and definition

1.1   Background

Today's planners do not meet the needs of a user. Person willing to easily schedule an event can stump into many problems in the process of organizing an event. The problem can become even more complex when the number of people willing to attend the event grows.

The proposed timeslots can be spread over few different event times, because each of the attendants have their own schedules they have to cope with and they have scheduled before. Because of the many constraints force upon the other attendances it is hard to find appropriate time slot that can work out for everyone. Usually in this situation the event is held over email conversation, which can become quite messy.

1.2   Description

To solve these problems, we need to create an event planner that will manage to merge the information from all attendances calendars and manage to find a free time slot that fits everyone. MeetMe planner should be able to draw information from applications such as Google Calendars to provide enough data for finding an appropriate time slot for the event. And the application should be able to propose a time slot by respecting the privacy and security of the shared attendances calendars i.e. with minimum disclosure of the attendance plans.

2. Actors

Visitor   A visitor is a potential user of the system which can perform the following use cases:
- Registration

User   The system's user can perform the following use cases:
- Login
- Logout
- Edit profile
- Add calendar
- Delete calendar
- Edit calendar
- Invite friends

- Recover password
- Synchronize calendar
- Run calculation manually
- See all events
- Edit event
- Delete event
- Schedule event

Attendant An attendant is a user who chose to not share his private calendar which can perform following use case:

--- Choose a time slot

3. Functional requirement

The following functional requirement describe the system's behavior with respect to JustMeet calendar.

3.1. Use cases

3.1.1. Visitor

3.1.1.1. Registration

Name	Registration
Actor	Visitor
Entry condition	The visitor does not already have an account
Flow of events	1. Visitor navigates to the registration page 2. Visitor fills the registration form 3. Visitor confirms registration 4. Systems add the new user to DB 5. Visitor receives a confirmation message of successful registration
Exit conditions	• Visitor's account is successfully store in DB Visitor is redirected to the login page
Exceptions	• Visitor enters already used email • Visitor enters wrong input • Visitor leaves compulsory fields empty • Visitor refreshes the page without submit all inputs

3.1.2. User

3.1.2.1. Login

Name	Login
Actor	User
Entry condition	● User must be registered and not logged in
Flow of events	1. User navigates to the Login page 2. User enters his email 3. User enters his password 4. User clicks Login button 5. System checks his credentials
Exit conditions	● User logged in and redirected to home page
Exceptions	• User enters a wrong credentials • User refreshes the page without submit all inputs User forget his password

3.1.2.2. Logout

Name	Logout
Actor	User
Entry condition	● User already logged in
Flow of events	1. User clicks on Logout
Exit conditions	● User redirected to login page
Exceptions	

3.1.2.3. Edit profile

Name	Edit profile
Actor	User
Entry condition	• User already logged in
Flow of events	1. User clicks "Personal info" 2. User updates settings 3. User clicks confirmation button 4. System updates the changed data in DB
Exit conditions	• User redirected to home page
Exceptions	• User enters wrong input • User leaves compulsory fields empty • User refreshes the page without submit all inputs

3.1.2.4. Add calendar

Name	Add calendar
Actor	User
Entry condition	• User already logged in • User has a calendar
Flow of events	1. User clicks on "Personal info" 2. User clicks on "Add calendar" button 3. User Choses between calendar's options 4. User enters his Google Calendar credentials, iCal URL or import iCal file 5. System stores the user's free slots from his calendars to DB
Exit conditions	• User redirected to home page
Exceptions	• User enters wrong credentials • User refreshes the page without confirming

3.1.2.5. Delete calendar

Name	Delete calendar
Actor	User
Entry condition	• User already logged in • User has Google Calendar account
Flow of events	1. User clicks on "Edit profile" button 2. User clicks on "Delete calendar" button 3. User confirms deletion of the calendar 4. System deletes calendar from the DB and updates user's table
Exit conditions	• User redirected to home page
Exceptions	• User refreshes the page without confirming

3.1.2.6. Edit calendar

Name	Edit calendar
Actor	User
Entry condition	- User already logged in - User has Google Calendar account
Flow of events	1. User clicks on "Personal info" 2. User clicks on "Add calendar" button 3. User Choses between calendar's options 4. User enters his Google Calendar credentials, URL or import calendar 5. System stores the user's free slots from his calendars to DB
Exit conditions	User redirected to home page
Exceptions	- User enters wrong credentials - User refreshes the page without confirming

3.1.2.7. Invite friends

Name	Invite friends
Actor	User
Entry condition	User already logged in
Flow of events	1. User clicks "Invite friends" button 2. User writes friend's email 3. User clicks invite button 4. System sends Invitation to join the application via email
Exit conditions	
Exceptions	- User writes a wrong email - Email already exists in DB

3.1.2.8. Recover password

Name	Recover password
Actor	User
Entry condition	User already registered to the system
Flow of events	1. User clicks on "forgot password" 2. User enters his email address 3. System sends a reset link to provided email 4. User clicks on the link 5. User enters his new password

	6. User re-enters his new password
	7. User clicks on "OK" button
	8. System updates the user password on the DB
Exit conditions	● User redirected to login page
Exceptions	● User enters inexistent email on the DB
	● User re-enters password different then the first

3.1.2.9. Synchronize calendar

Name	Synchronize calendar
Actor	User
Entry condition	● User already logged in
	● User already added his calendar
Flow of events	1. User clicks on "Edit profile"
	2. User clicks on "Synchronize calendar" button
	3. System connects to the user calendar
	4. System updates calendar data to DB
Exit conditions	● User refreshes the home page
Exceptions	

3.1.2.10. Run calculation manually

Name	Run calculation manually
Actor	User
Entry condition	● User already logged in
	● User is the owner of the event
Flow of events	1. User selects the event
	2. User clicks on "Run calculation" button
	3. System synchronizes all attendant calendar
	4. System handles time zones
	5. System calculate the best event time
	6. System stores the event time on the DB
	7. System notifies attendant about the event time by email
Exit conditions	● User redirected to home page
Exceptions	● User refreshes the home page without confirming

3.1.2.11. See all events

Name	See all events

Actor	User
Entry condition	● User already logged in
Flow of events	1. System selects user's events from DB 2. System shows events to user on the home page
Exit conditions	
Exceptions	

3.1.2.12. Edit event

Name	Edit event
Actor	User
Entry condition	● User already logged in ● User already the owner of the event
Flow of events	1. User clicks on my events 2. Systems shows all the events created by the user 3. User clicks on edit icon 4. User updates the event 5. User enters Minutes of Events during the events 6. User confirms his choices 7. System updates the event on DB 8. System sends an email for event attendants
Exit conditions	● User redirected to home page
Exceptions	● User enters wrong input ● User leaves compulsory fields empty ● User refreshes the page without submit all inputs

3.1.2.13. Delete event

Name	Delete event
Actor	User
Entry condition	● User already logged in ● User already the owner of the event
Flow of events	1. User clicks on my events 2. Systems shows all the events created by the user 3. User clicks on edit icon 4. User updates the event 5. User confirms his choices
Exit conditions	● User redirected to home page
Exceptions	● User refreshes the page without confirm deletion

3.1.2.14. Schedule event

Name	Schedule event

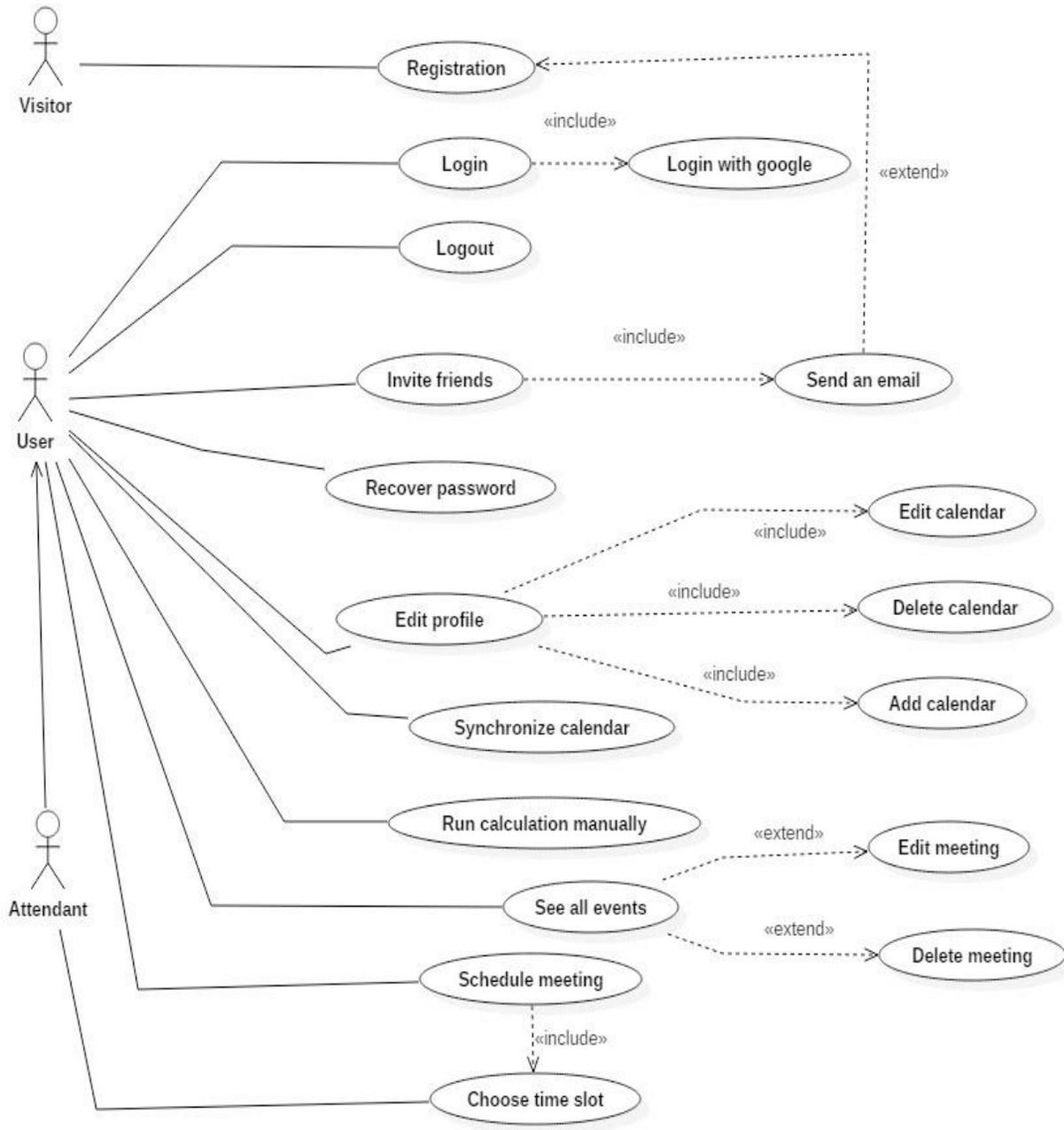| Actor	User
Entry condition	● User already logged in
Flow of events	1. User clicks "Schedule an event" button 2. User enters event's title 3. User enters n's description 4. User enters the things he most do before event 5. User enters event duration 6. User enters the preferred time slot 7. User enters tag to event 8. User enters email of attendants 9. User clicks "Create" button 10. Systems adds the event to DB 11. System sends email to attendant with event's information and link to event's page 12. User clicks on the link provided by the email 13. Systems shows a login page 14. User without account can register to the system or skip the page 15. User without account enters their email and selects the time slots they desired 16. User confirms his choices 17. User with account login to the system 16. User selects desired time slots
Exit conditions	● User redirected to home page ● User redirected the thanks page
Exceptions	• User leaves compulsory fields empty • User refreshes the page without submit all inputs

3.1.3. Attendant
3.1.3.1. Choose time slot

Name	Choose time slot
Actor	Attendant
Entry condition	● User already registered to the system ● User chooses not to share his personal calendar
Flow of events	1. User opens the event link 2. User selects desired time zone 3. User selects the time slots he desired 4. User clicks on "Confirm" button
Exit conditions	● User redirected to home page
Exceptions	● User refreshes the page without confirming

3.2. Use cases diagram

The following use cases diagram represent graphical representation based on the User Cases mentioned above.

4. Non Functional Requirements

4.1. Performance Requirements

Performance of the system should be good enough to provide the user with fast responding software system. Response time should be small enough to enable good user experience.

4.2. Software System Attributes

- Availability, the application should be available to handle user's request at all times using any device with an installed web browser.
- Maintainability, the application used known design patterns in order to provide easier updating of system with new features.
- Portability, the application could be acceded by mobile devices on web browser.
- User Interface, the web application should be intuitive so even the nontechnical users can use the system as simply and efficiently as possible.
- Security, application will not take advantage of user's personal information and will respect the privacy policy related to information enclosed in his Google calendar account.

5. User stories

User stories are short and simple sentences that contain the features that customer expects to be present in the system. Product owner is responsible for them: team should create a system that meets the customer's expectations using user stories. The customer's requirements aren't equally important; for this reason high, average or low priority is attributed to each of them.

ID	USER STORY	PRIORITY	Linked use case
UserStory 1	As a visitor I want to be able to register to the system ● The user should be able to get in the web application and register in the network	High	Registration

UserStory 2	As an user I want to be able to login/logout at the application with my account ● The user should be able to login and logout of the web application whenever he wants to	High	Login Logout
UserStory 3	As an user I want to receive an e---mail with my new password if I forget the first one ● Whenever the user forget this password, the application should provide an way to allow the user recover his password	High	Recover password
UserStory 4	As an user I want to schedule an event or an event ● The user should be able to plan an event by saying the its name, time slot and the attendants	High	Schedule event
UserStory 5	As an user I want to Edit profile ● All the users should have the option to change all the settings, for instance, change password, change profile picture, change email, change nickname and other privacy settings or Add/Edit/Delete calendars	Average	Edit profile Add Calendar Edit Calendar Delete Calendar
UserStory 6	As an user I want to receive notification when an event is scheduled ● The user should receive an notification by email whenever someone wants him as an attendant in an event	High	Schedule event
UserStory 7	As an user I want to invite my friends to use the application ● The application should provide some way to allow the users to invite other people to join the community by sending an email to user's friends.	Average	Invite friends
UserStory 8	As attendant I want to able to choose right time slot from me without sharing my calendar ● Whenever some user is marked as an attendant in some event/event, the application should check on user's calendar for the best time slot, but the privacy of the user must be respected. So no information of the user calendar can be shared with the other users.	High	Choose time slot

UserStory9	I want to be able to login using my Google account • We need to provide option for user to login with his Google account and automatically connect to his Google calendar	Low	Login with Google
UserStory10	User calendars must be up to date • Use different options to always keep users calendars up to date	High	Synchronize calendar
UserStory11	As an User I want to be able to get potential event time • Provide option to manually and automatically run calculation of best event time but also allow user (event organizer) to decide about best option	High	Run Calculation manually
UserStory12	As an User I want to check all events • We will provide option to see, edit and delete past and future events	Average	See all events Edit event Delete event

5.1. Sub user stories

UserStory 4		
UserStory41	As an user I want to choose attendants in the event or event ● The user should be able to choose the attendants of the event by writing the email, nickname or real name	High
UserStory42	As an user I want to be able to receive event notifications ● Whenever an event as some updates like new attendants, new time slots or other details , all the attendants must be informed with an email	Average
UserStory43	As an user I want to choose attendance mandatory ● The user that schedules an event should be able to choose if some user is mandatory to the event or not	Average
UserStory44	As an user I want to obtain the best time for the event using my and the other participants information in Google Calendar ● The application must check the user's Google Calendar to check if him doesn't have some overlap event in that time slot	High

UserStory 5		
UserStory51	As an user I want to be able to edit my profile info • User should be able to change Name, surname, password, profile picture etc.	Average
UserStory52	As an user I want to be able to Add calendar • User should be able to link his Google Calendar to our app	High
UserStory53	As an user I want to be able to Edit my calendar • User should be able to change his Google Calendar	Average
UserStory54	As an user I want to be able to Delete my calendar • User should be able to remove his calendar from account. Calendar must be deleted from database.	Average

UserStory 8		
UserStory81	As an user I want to distinguish between preferred event times and possible events times • The user should be able to say which time slots are better for him, because sometimes we have availability in some time slot but we prefer another.	Low

UserStory 10		
UserStory101	As an user I want to manually synchronize my calendar • There should be option for user (next to calendar info) for manually synchronization	Low
UserStory102	Automatically run synchronization before running calculation • Before calculation of best event times synchronize calendars for all attendants	High
UserStory103	Automatically run synchronization in specific intervals • Use interval of every 12h to automatically synchronize calendars of all users	Low

UserStory 11

UserStory111	As an user I want to manually run calculation & choose best option • When user choose to manually run calculation of potential event times, system will get all calendars from attendants who shared their calendars and perform calculation (While taking care of time zones). As output, system will provide top X of possible event times, while user chooses best option for him. On the end notify all attendants of chosen event time.	High
UserStory112	Automatically run event time calculation • When last attendant shares his calendar run calculation. Notify only event organizer so he can decide between possible event times.	Average

UserStory 12

UserStory121	As an User I want to see my events • User will have option to see events which he was attending and he is attending	Average
UserStory122	As an User I want to edit my event • User will have option to edit his event	Average
UserStory123	As an User I want to delete event • User will have option to delete his event	Low