Use-Cases Specifications Document (UCSD)

Acronyms			
UAB UPF	Universitat Autonoma de Barcelona Universitat Pompeu Fabra	SE	Software Engineering
NA	Neighbourhood association	PA	Property Administrator
DB	DataBase		

Revision History			
Version	**Date**	**Comments**	**Autor**
0.1	1-Apt-2019	Initial Draft	ES Team
0.2	26-May-2019	Second version	ES Team
0.3	2-June-2019	Third version	ES Team

Contents

List of Figures

List of Tables

1 Introduction

This document describes the most relevant use cases for the MY-NEIGH software.

A use case specifies the system behavior of the system and represent the functional requirements of the system. They are started by a user with an objective and they are completed when the system satisfies this objective.

In order to obtain a use case, we should first identify the actors, find all the roles that can be played by the users, identify the objectives to be fullfilled, create a Use Case per object and finally, obtain a Use Case Diagram.

Regarding the relationships between Use Cases, there are three: generalization, a Use Case inherits the behavior of another, inclusion, which incorporates explicitly the behavior of another Use Case at some point and extension, which a Use Case completes the funcionality with another user.

This document starts with an Introduction. Next, a Use Case diagram is presented and some Use Cases are specified. Each Use Case specification has a description with the following elements:

- Basic flow
- Alternative flow
- Pre-Conditions
- Post-Conditions

2 Use Case Diagram

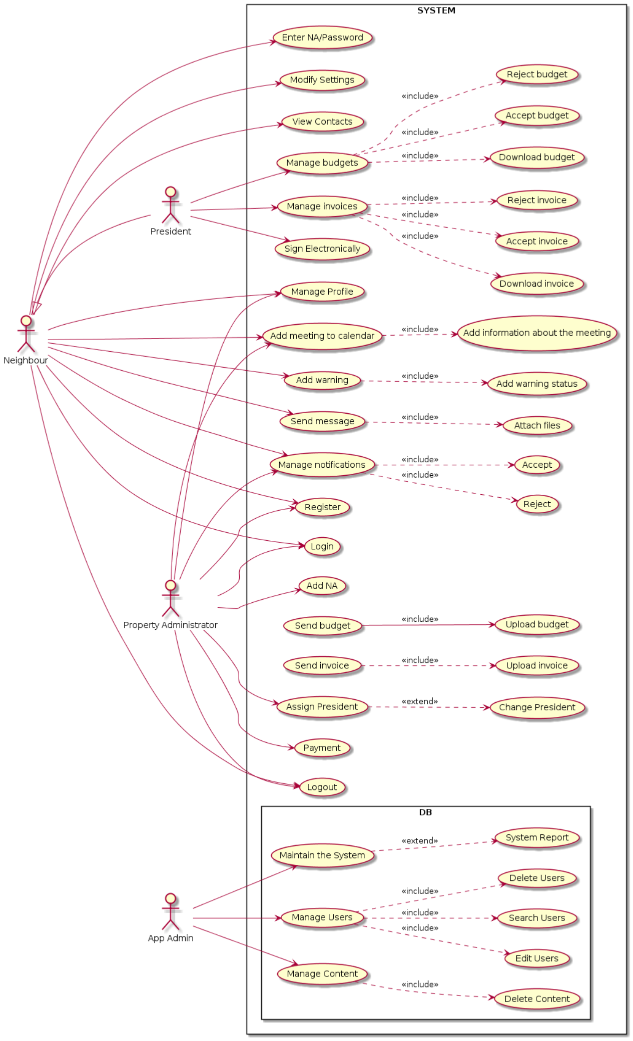

Figure 1: Total Use Case Diagram

2.1 Specific Use Case Diagrams

2.1.1 Neighbour

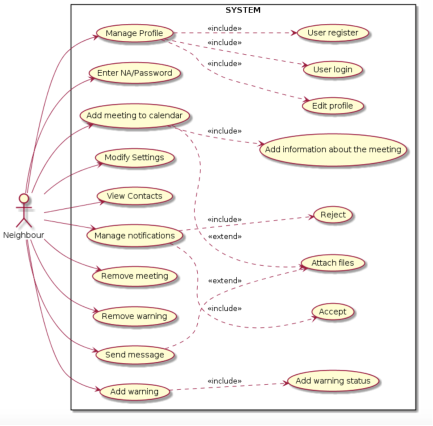

Figure 2: Neighbour Use Case Diagram

2.1.2 President

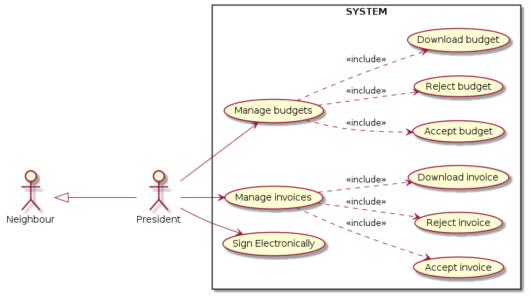

Figure 3: President Use Case Diagram

2.1.3 Property Administrator

2.1.4 APP Administrator

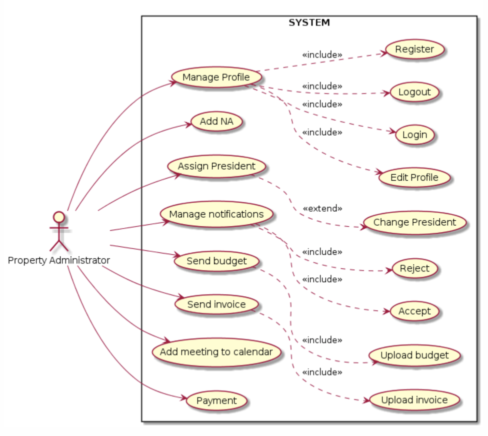

Figure 4: Property Administrator Use Case Diagram

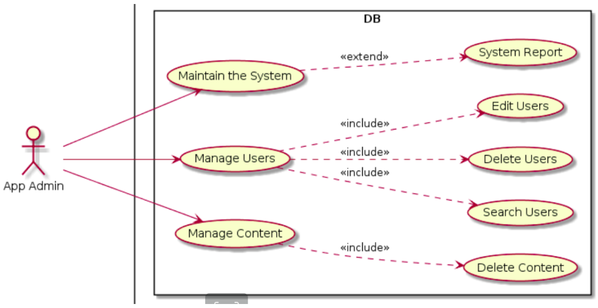

Figure 5: APP Administrator Use Case Diagram

3 Use Case : Add event to calendar

This use case consists on the user (neighbour, president or property administrator) to be able to schedule a meeting or an activity and add it to the calendar to everyone from the neighbor association can see it.

3.1 Events flow

3.1.1 Basic flow

The use case starts when the user has access to the system and is logged.

1. User must access to the calendar.
2. User must select a day, a month and a year from the calendar.
3. User must indicate it is going to add an event by clicking at the ”New Event” icon.
4. System must take information about the event. Required information:

   For *Neighbours*:

   - Name of the event.
   - Place of the event.
   - Time of the event.
   - Brief description of the event.
   - Priority (High, Medium or low)

   For *Property Administrators*:

   - Name of the event.
   - Place of the event.
   - Time of the event.
   - Day’s Order.
5. User must indicate confirmation of the event by clicking ”Okay” once all the information is covered.
6. System sends to all interested users a notification.
7. Users invited must accept or decline the invitation to the event by clicking ”Accept” or ”Decline”.
8. System shows confirmed users going to the event once the invitation has been accepted.
9. Use Case Ends.

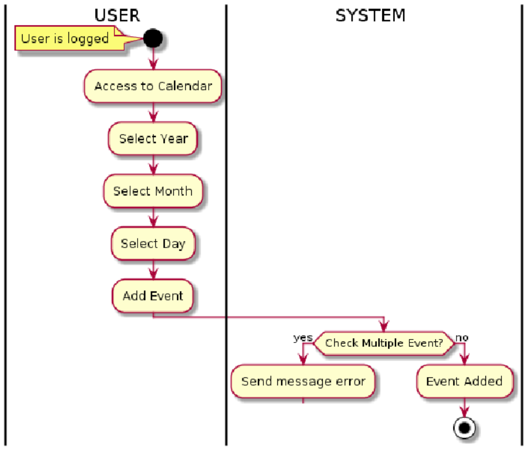

Figure 6: Activity Diagram of event addition

3.1.2 Alternative flow

- Cancel Event
  1. User selects cancel after clicking "Okay".
  2. User return to the calendar.
- Multiple Events.
  1. The system will display "Multiple Events at the same time".
  2. The system will keep none of the events.

3.2 Pre-Conditions

- User Registration
- User Login
- Be in an neighbourhood association

3.3 Post-Conditions

- The new event or meeting is added to the calendar.
- System shall send a reminder one day before the event depending on priority.

4 Use Case : Remove event from calendar

The actors involved in this use case are: neighbour, president and the property administrator. They need to be able to remove a meeting or an activity from the calendar.

4.1 Events flow

4.1.1 Basic flow

The use case starts when the user has access to the system and is logged.

1. User must access to the calendar.
2. User must select a day, a month and a year from the calendar.
3. User must select an Event that is already in the calendar.
4. User must click ”Delete Event”.
5. The system will remove the Event and the users will not be able to see it.
6. Use Case Ends.

4.1.2 Alternative flow

- Move Event
  1. User selects another day to introduce the event.
  2. User return to the main page.
- Delete Error.
  1. The system will display an error message ”The Meeting cannot be deleted”. This can happen when a neighbour tries to delete a meeting added by the property administrator.
  2. The meeting will remain in the calendar.

4.2 Pre-Conditions

- User Registration
- User Login
- Be in an neighbourhood association

4.3 Post-Conditions

- The meeting is removed from the calendar.

5 Use Case : Edit event from calendar

The actors involved in this use case are: neighbour, president and the property administrator. They need to be able to edit meeting or an activity information from the calendar.

5.1 Events flow

5.1.1 Basic flow

The use case starts when the user has access to the system and is logged.

1. User must access to the calendar.
2. User must select a day, a month and a year from the calendar.
3. User must select an Event that is already in the calendar.
4. User must click "Edit Event".
5. User must change or introduce new information.
6. The information will be updated.
7. Use Case Ends.

5.1.2 Alternative flow

- Invalid Information.
  1. The system will display an error message "Invalid Information".
  2. The user will need to re-enter the information to the calendar.

5.2 Pre-Conditions

- User Registration
- User Login
- Be in an neighbourhood association

5.3 Post-Conditions

- The informatio of meeting is updated.
- The system will send a notifcation to other users with updated information.
- Users must accept or decline the invitation to the event by clicking "Accept" or "Decline".
- System shall send a reminder one day before the event depending on priority.

6 Use Case : User management I - Delete User

An administrator deletes an existing user in the database.

6.1 Events flow

6.1.1 Basic flow

This use case starts when an administrator enters to the system.

1. The system displays the list of users.
2. The administrator selects the user that wants to delete.
3. The system show the information of the user.
4. The system removes the user selected from the database.
5. The system database is updated.

6.1.2 Alternative flow

- Cancel deletion. Administrator can cancel the action mentioned, while the remove option is not selected.
- Deletion Invalid. Administrator should update the database.

6.2 Pre-Conditions

- Administrator should have access to the system.
- Data Base control manager shall be in operative state.

6.3 Post-Conditions

- Administrator may send a notification to the deleted user.
- User will not be in the database.

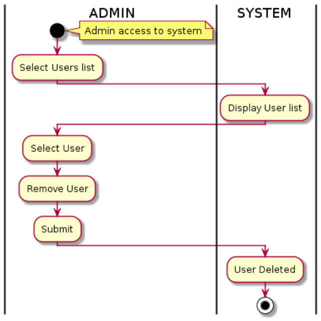

Figure 7: Activity Diagram of user deletion

7 Use Case : User management II - Modify user

An administrator modifies the profile of an existing user in the database.

7.1 Events flow

7.1.1 Basic flow

This use case starts when an administrator enters to the system.

1. The system displays the list of users.
2. The administrator selects the user that wants to modify.
3. The system show the information of the user.
4. Administrator submit the user profile modification.
5. The system updates the profile information.

7.1.2 Alternative flow

- Cancel modification. Administrator can cancel the action mentioned, while the submit option is not selected.
- Invalid Modification. An error will appear and the administrator should enter again the information.

7.2 Pre-Conditions

- Administrator should have access to the system.
- Data Base control manager shall be in operative state.

7.3 Post-Conditions

- The new information is in the user profile.

8 Use Case : User management III - Search User

An administrator search an existing user in the database.

8.1 Events flow

8.1.1 Basic flow

This use case starts when an administrator enters to the system.

1. The system displays the search box.
2. The administrator enters the string or criteria.
3. The system shows the list of the user that follow a determined criteria.

8.1.2 Alternative flow

- Cancel search. Administrator returns to the main page.
- Invalid Enter. An error will appear and the administrator should enter again the information.
- No founds. There are no results.

8.2 Pre-Conditions

- Administrator should have access to the system.
- Data Base control manager shall be in operative state.

8.3 Post-Conditions

- The new information is the user profile.

9 Use Case : Manage Notifications

The different users can accept or reject the different types of notifications that they receive.

The actors involved in this use case are: neighbour, the president and the property administrator.

9.1 Events flow

9.1.1 Basic flow

For the *neighbour*:

This use case starts when the neighbour is logged into the system.

1. The system shall show a notification of an event added.
2. In the notification there will be a button to click for the attendance to a meeting created either by the president or the property administrator.
3. The neighbour shall choose to accept or reject the attendance to the meeting.

For the ***president***:

This use case starts when the president is logged into the system.

1. After the administrator has send a budget or an invoice.
2. The system shall send a notification to the president.
3. The president shall open the finance section.
4. Use Case Ends

For the *property administrator*:

This use case starts when the president is logged into the system.

1. The system shall show a notification for the acceptance or rejection of the budget of the community.
2. The property administration shall see if there are some notes from the president.

9.1.2 Alternative flow

- Remember later.
    1. The user can choose to receive the notification after a period of time.
        - 5 minutes
        - 1 hour
        - 1 day

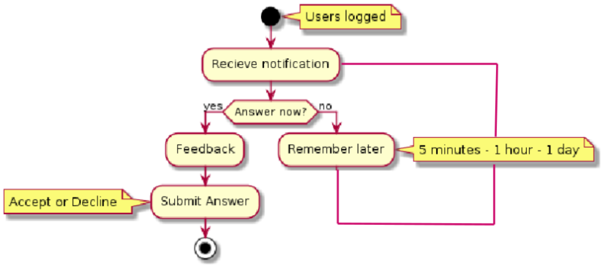

Figure 8: Notification management Activity Diagram

9.2 Pre-Conditions

- User shall have access to the system.

9.3 Post-Conditions

For the ***president***:

- The user will be in the placed in the list of attendency, if it accepts.

For the *president*:

- The president will be able to download the budget or invoice.

For the *property administrator*:

- The PA will see if the President has accepted or not the budget or invoice.