

# SELFPLANNER 2.0: Storyboard

Ioannis Refanidis<sup>1</sup>, Anastasios Alexiadis<sup>1</sup> and Neil Yorke-Smith<sup>2</sup>

<sup>1</sup>University of Macedonia, Dept. of Applied Informatics, Thessaloniki, Greece  
{yrefanid, talex}@uom.gr

<sup>2</sup>American University of Beirut, Lebanon and SRI International, US  
nysmith@aub.edu.lb

## Abstract

This text describes the life presentation of SELFPLANNER 2.0 at ICAPS-2011.

## Introduction

SELFPLANNER 2.0 is a deployed system available at <http://selfplanner.uom.gr>, currently being used by some dozens of users. So, users can simply register with the system and start playing with it themselves. Using the system a user can:

- Define new activities. Each activity is characterized by its temporal domain, the minimum and the maximum possible duration, a set of locations, its utility, a utilization value denoting whether the activity can be scheduled simultaneously with other activities or not and a flag denoting whether the activity is interruptible or not.
- In case of interruptible activities, additional constraints can be defined over the way the parts of the activity are scheduled. Particularly, the user can define the minimum and the maximum duration of any part of the activity, as well as the minimum and the maximum temporal distance between pairs of parts of the activities. Binary constraints can be either hard or soft. In the latter case, the user can define a utility value that is obtained in case of satisfaction. Partial satisfaction of soft constraints is allowed.
- Define constraints between pairs of activities. Ordering, proximity and implication constraints are supported. Again, constraints can be either hard or soft. In the latter case, the user can define a utility value that is obtained in case of satisfaction. Partial satisfaction of soft constraints is allowed.

- Select the Google calendar where its activity will appear when scheduled.

## Scenario

Our storyboard consists of demonstrating the functionalities of the system by inserting, in real-time, several activities into the user's calendar using SELFPLANNER 2.0. Initially, the user's calendar consists of activities with precise time and space reference, i.e., they cannot be rescheduled. For example, these activities may concern classes or meetings with other people (remind that meeting arrangement is not supported by SELFPLANNER).

In the following we present four such cases.

### Case 1: Getting the children to the school

Each morning (excluding weekends) the user needs to get the children to the school. This takes place precisely at 08:30. From the school, the user needs 75 minutes to go to his work using his car. The activity is periodic, with a constant time and location reference.

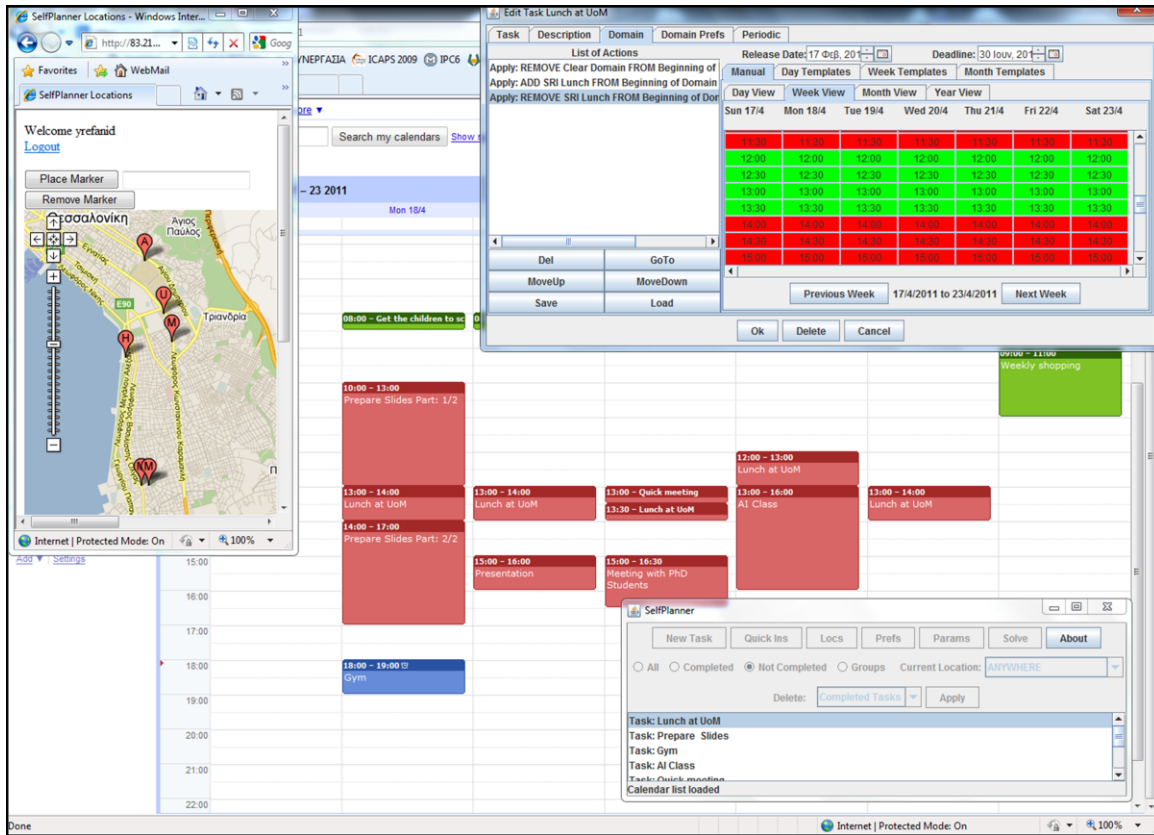
### Case 2: Going to the mall

The user enters a simple activity with constant duration, which has to be performed in a specific place, e.g., the local mall. The activity is non-interruptible, it has a constant duration of two hours (excluding time devoted to transportation), and can be performed only when the mall is open. The activity is also periodic, with a weekly period, and is preferred to be scheduled as late as possible within a week, e.g., on Saturday.

The activity is inserted with the above details and is scheduled on Saturday.

### Case 3: Lunch break a work

Each day the user has a lunch break at work. The restaurant at work is open between 12:00 and 14:00. The user prefers to spend one hour for the lunch, however she is also



**Figure 1.** An overview of the SELFPLANNER system. The main application window (lower right) contains the current list of activities. In the upper right corner is the Edit Activity dialog box, shown editing the temporal domain of the Lunch at UoM activity. It is interesting to have a look at how this activity has been scheduled. It is a daily periodic activity with a duration that may be either 30 mins or 1 hour (preferred). The restaurant is open from 12:00 to 14:00 daily, and the user prefers to eat as late as possible, i.e., close to 14:00, more strongly than she prefers to have 1 hour for lunch. Seen on the left is the interface for designating relevant locations. Behind the other windows is the Google Calendar interface in which the whole schedule is shown.

satisfied with half an hour. He prefers to have the lunch as late as possible, and this preference is stronger than having a whole hour for the lunch.

The activity is inserted as a daily periodic activity (excluding weekends), with the aforementioned details. SELFPLANNER schedules the activity each day, according to the user's preferences, i.e., at 13:00 and with duration equal to a whole hour. However, there are two exceptions: On Thursday the user has a class at 13:00, so the lunch break is scheduled from 12:00 to 13:00. Furthermore, on Wednesday, the user has a quick meeting from 13:00 to 13:30. So, there are two options: Either scheduling the lunch break from 12:00 to 13:00 or scheduling the lunch break from 13:30 to 14:00. Since the user is mostly interested in having his lunch as late as possible, than having a whole hour for his lunch, the latter option is selected.

#### Case 4: Preparing slides

The user has to prepare his slides for a class. He estimates that he needs six (6) hours to prepare the slides. This time may be split in parts, i.e., the activity is interruptible. The user prefers not to split the activity in parts smaller than two hours, whereas he is ok with scheduling the whole duration of the activity together. The activity can be scheduled within working hours only. Finally, the user prefers to schedule the activity as early as possible within a week.

The user enters all these data into SELFPLANNER 2.0. The activity is scheduled in two parts, with three hours duration each. One is scheduled in Monday's morning and the other in Monday's evening, with the lunch break separating them in time.

Figure 1 gives a snapshot of the user's calendar after inserting the aforementioned activities. Up to ICAPS-2011, we intend to devise more such scenarios, demonstrating all aspects of the system.