Smart Commute

**What is it?**

Smart Commute is a web based application that serves as a tool for the busy day-to-day user. It’s essentially an event manager that also aids the user visually by showing any scheduled appointment or event on the map, as well as how to move from one scheduled event to the other.

**Who will enjoy using Smart Commute?**

This app is dedicated to anyone who has a highly dynamic everyday life: the executive moving from meeting to meeting in different locations, the father/mother of three picking up her children, the freelancer photographer/designer with a high amount of photo-shoots/auditions, job-seeker assisting interviews, college student attending group meetings, anything that requires activity within several locations in a limited amount of time.

**Why would you like to use Smart Commute?**

The backbone objective of this application is to facilitate the day organization considering the transport means used by the user; his/her scheduled events, and the available time to go from event to event considering the time it takes to go there. It is fully oriented to simplify the set of tasks during the day and preventing uncertainties as much as possible. Very often, it is necessary to consult more than one application while organizing the day: maybe a digital agenda is necessary to check that dentist appointment, but after that there was a scheduled lunch with a coworker which wasn’t specified anywhere (however the location and time was arranged), and lastly a very important meeting reunion with the boss which was set in the default calendar app of the user’s PC or Smartphone. On top of that, once the next event time is approaching, you must be sure to know what your next task is, when to leave and where to go to.

Smart Commute does all. It just needs you and your appointments! No need to switch apps or tabs. You say what your events are, what transport you prefer to use, and the rest is done for you. Additionally, it handles your city’s transport setbacks, strikes, or any unexpected unavailability, chooses the alternative, and lets you know. Smart Commute is the clean & neat agenda and assistant.

**What do you need to use Smart Commute?**

All you need is a browser, in whichever device you’re in. Supports all browsers and pretty much any device can access Smart Commute without crashing.

**What drives us to create Smart Commute?**

We know that you out there are quite busy and can’t choose what utility app will help you the best, or are annoyed by using the default/standard/traditional ways. Take the leap with us: all in one app for the dynamic worker that hasn’t found the ultimate solution? Think of Smart Commute: a utility that attempts to fuse where and when seamlessly, and serves it to you in a silver plate.

Project objective

* Develop a scheduler web-app with a calendar interface that computes and accounts for travel time between appointments to make sure you’re never late for an appointment.

Functionality

### Authentication System

Allow access per user to its own private environment through login authentication, and new users to register into the system.

### Calendar interface

Let the user interact with a calendar interface and manage appointments (events) within it, by performing the following actions: create, edit, delete and save. Additionally, it will allow switching between daily, weekly and monthly view, as well as schedule mode, which lists all the events saved. It will be possible to switch months aside from the current one, as well as weeks and days just like any other standard calendar. Creating an appointment will generate a node in the roadmap interface and once following events within a day are created, the app will calculate the time required to move (according to user’s preferences) from one event to the other and approve the following event creation if there’s available time to commute, or deny/warn the event creation otherwise.

### Smart Commuting

Whenever two or more events are generated by the user within a day, the app will automatically compute the travel time between appointments and warn for time overlapping. Smart Commute is designed to warn the user minutes before needing to depart to the next event if the available time to get there is running short. The app has a built-in map interface powered by Google Maps where the event markers of the day’s schedule will be shown, its details, and the travel directions from event to event, including their duration.

### Travel means by appointment and by day

The web app must allow the user to specify the preferred travel means and to be weekday sensitive, as well as enabling or disabling transport means available, which are: bus, car, tram, metro, train, bike or walk. Smart Commute will automatically compute the commuting time between appointments by using the preferred transport, and if not available, it will use the fastest alternative the user allows. There is also an “auto” option which doesn’t specify any user preference but instead prioritizes transport depending on its duration (lower is better).

### User customizable

Let the user configure its workdays (what days of the week Smart Commute will assist the user). This versatility was thought since it’s very likely that not all users have a standard Monday-Friday schedule: some can work on Saturdays, or some may have a weekday free.

Allow modifying predefined breaks: there are “break times” we already set up for you. These are daily events with no location, just a time gap specified by the user in which no event can be scheduled, and no travel is considered. It will also allow the user to change per day the routine according to the daily requirements, if need be.

The app will also allow the modification of alert times. By default, the app notifies the user 15 minutes before the minimum time allowed to commute is reached but any alert time (up to 25 minutes) can be configured, including disabling the feature.

### Variability time between appointments

Allow the user to modify on the go the event durations. Because being fully organized is almost impossible, this feature exists. The app will allow the user to quickly extend event durations and recalculate the schedule, and warn back if time violations are met.

# Specification

The user, by providing its name, last name, email and a password, will be able to register and create an account within the app. The system will then validate the login credentials and if successful it will connect the user into its account.

The user will have the possibility to create, edit, delete and save as many appointments as it wants. Each appointment will have the following fields: title, date, start time, end time, address and description.

The calendar interface will provide day, week, month and yearly views, and the addition of events can be accessed through any of these views. Also, a schedule view is available which lists all the events within a time period.

Moreover, a roadmap view will be available. This interface shows the city map with the day roadmap of the user: all the current day’s events will appear as nodes, and upon clicking given nodes the appointment details will be shown. Additionally, the travel trajectory between appointments (nodes) will appear as calculated from the Google Maps API with the preferred transport, and by clicking on it, its details (duration, transport to be used and maximum departure time) will appear.

The travel time (duration) before mentioned will be computed and alerts will be generated whenever the maximum departure time is near. Additionally, suggests the preferred transport as main option and discards the transports disabled by the user.

Through the settings interface the user will be able to personalize its preferences, which will be the selection of the preferred transport, toggling between available transports (bike, walk, car, bus, tram, metro and train), specifying resting times (by default lunch and break times) and setting up the alerts.

# Description

In the application, each user logs into its account by entering its email and password, the system validates the login and if successful the system connects the user into its account. If not successful, the system prevents entry but allows repeated attempts. After having validated the credentials, the user is redirected to the calendar interface automatically and if the user had previously saved a configuration and events, they will automatically load.

Once the user is in the calendar interface, it will have the possibility to take four main actions: to manage/view appointments, to configure settings, to view roadmap, and to logout. Managing appointments, configuring settings and viewing the roadmap are interfaces which by clicking their link, can be toggled among each other.

The functions of viewing appointments are within the same calendar interface. The events are automatically shown in the calendar interface, where it is possible to see the monthly view as default and the user can toggle views to daily, weekly, yearly, and schedule view. Differently from the date-based views, the schedule view lists all the appointments sorted in order of how recent.

Adding an appointment can be done via clicking the “+” button, and a configuration window will appear where the user can input the title, date, start time, end time, address and description of the event. The fields of title, start time, end time and date are mandatory, and description and address are optional. Once user has finished, “cancel” and “save & exit” options are available to save the event or to discard changes.

To edit/delete any shown appointment, the user can double click the shown event in any of the interface views (except schedule view) and the same fields as the create event window will appear, showing the previous event configuration, in addition to a “delete” button.

To configure settings, the user clicks the settings tab from the calendar interface and once its redirected to the settings interface it has the possibility to modify the preferred transport, checking which transports to use (by toggling enable/disable buttons), specifying the resting times (which by default are called lunch and break times), and specifying the alerts.

Selecting the preferred transport will give priority to that type of transport, so whenever it is available once the Google maps API calculates the trajectory the roadmap interface will automatically show that given trajectory with the specified transport type. If the preferred transport is not available, it will prompt the user and ask for an alternative and show the fastest as recommended.

Enabling/disabling available transport types will force Google maps API to not show certain transports. For example, if you don’t desire to use tram and disable the option, under the scenario that your preferred transport isn’t available, even if for that travel tram is the recommended way, it will not be shown.

Specifying the resting times is a functionality that allows the user to have a daily fixed period (also configurable to any preferred days of the week) reserved to rests. Rests are considered a repetitive weekly event, and within the event period no events can be added, and travel time is not supposed to overlap that period either. By default, the resting times are named lunch time and break time. By clicking the edit rest times button the user will decide for each of the rest times their names, day periods, and which day of the week they should be active. Deselecting all days disables the resting time.

The alerts section decides how close to the maximum departure time (to the next appointment) should Smart Commute notify you. By default it is set to fifteen minutes.

The roadmap is available through the button roadmap, present in the calendar and settings interfaces. The roadmap shows by default the Google Maps API centering your location and it loads any nodes (events) and edges (travels) previously computed, if any. This functionality allows the user to view the “roadmap” of its scheduled day according to the events it created. It shows the appointments as pins in the map that are clickable. Once clicked, they show the event details. Additionally, the “edges” which are the travel distances, are computed automatically thanks to the Google Maps API and shown accordingly, with a dialog box specifying which transport type is being used, and the duration of the commuting time.

Logging out is a button available within the roadmap and calendar interfaces, and by clicking on it the user logs out, the app saves the generated data and the configuration data to the server, and it prompts back to the login site.