

## **Project TimeSafe**

### **/TimeSafe/OperScripts**

Operational Scripts

Version: 1.0

Printed by: Artem Danielov

Printed on: 02 June 2002

# Contents

<b>1</b>	<b>Purpose of the document</b>	<b>1</b>
<b>2</b>	<b>Creating new task</b>	<b>1</b>
2.1	New short task	1
2.2	New long task	1
2.3	New long taskwith pre-task	1
2.4	New complex task	1
2.5	Task depending on other tasks	2
2.6	New subtask for a complex task	2
2.7	Repeating task	2
2.8	Long tasksplit into parts	2
<b>3</b>	<b>Task collisions</b>	<b>3</b>
3.1	No time for the new long task	3
3.2	Short task colliding another short task	3
3.3	Short task overlapping long task	3
3.4	Long taskoverlapping long task(s)	3
3.5	Long taskoverlapping short task(s)	3
<b>4</b>	<b>Organizing tasks</b>	<b>3</b>
4.1	Scheduling tasks for the nearest day	4
4.2	Scheduling tasks for the nearest week or month	4
4.3	Identifying taskoverlaps	4
4.4	Splitting a long task	4
4.5	New part of a long task	4
<b>5</b>	<b>Reminder</b>	<b>4</b>
5.1	Accepted task	4
5.2	Postponed task	4
5.3	No response from user	4
<b>6</b>	<b>Performing a task</b>	<b>4</b>
6.1	Starting on reminder	4
6.2	Starting without reminder	5
6.3	Pausing	5
6.4	Stopping	5
6.5	Completing	5
<b>7</b>	<b>Setting alarm clock</b>	<b>5</b>
7.1	Default alarm settings	5
7.2	Changing alarm settings for particular day	5
<b>8</b>	<b>Alarm clock activation</b>	<b>5</b>
8.1	Usual alarm	5
8.2	Launching application	6
<b>9</b>	<b>Synchronization</b>	<b>6</b>
9.1	PC with PC	6
9.2	CellCo with PC automatic on connection	6
9.3	CellCo with PC automatic without connection	6
9.4	CellCo with PC manual	6

## 1 Purpose of the document

The purpose of the operational scripts is to demonstrate main capabilities and main aspects of using TimeSafe. Operational scripts should help in defining and understanding system requirements. Operational scripts do NOT define exact program logic, exact user actions or exact user interface details.

## 2 Creating new task

### 2.1 New short task

User initiates creation of a new short task.

User enters **title** of the task.

User enters **detailed description** of the task, if needed.

User setsplanned **date and time**, when the task must be done, either exactly or as a time interval.

User sets **importance** of the task.

User setsdate and time of **deadline** for the task, if needed.

User enters **reason** for the **deadline**, if needed.

User selects **category** of the task.

User **saves** the task.

### 2.2 New long task

User initiates creation of a new long task.

User enters **title** of the task.

User enters **detailed description** of the task, if needed.

User sets **date and time**, when the task must start, either exactly or as a time interval.

User setsthe **time (duration)** required to complete the task – exactly or within some limits.

User sets **importance** of the task.

User setsdate and time of **deadline** for the task, if needed.

User enters **reason** for the **deadline**, if needed.

User selects **category** of the task.

User selects whether the task can be interrupted, **interruptions** are undesirable, or interruptions are impossible.

User selects whether the task can be **split into parts** or the taskmust be done completely at once.

User selects whether performing the taskrequires “**immersion**”.

*Immersion means process of focusing attention on the task. It takes usually 10-20 minutes.*

User **saves** the task.

### 2.3 New long task with pre-task

#### **Similar for post-task**

User makes all steps of “New long task” script before saving the task.

User enters **title** for the pre-task.

User enters the **time (duration)** required to complete the pre-task.

User selects whether the pre-task can be interrupted, **interruptions** are undesirable, or interruptions are impossible.

### 2.4 New complex task

User initiates creation of a new complex task.

User enters **title** of the task.

User enters **detailed description** of the task, if needed.

User sets **importance** of the task.

User sets **date** and time of **deadline** for the task, if needed.

User enters **reason** for the **deadline**, if needed.

User selects **category** of the task.

User **saves** the task.

User creates **subtasks** for the task. See “New subtask for a complex task” script.

## 2.5 Task depending on other tasks

*Dependency means that the task(dependant) cannot start until all foregoing tasks have been completed.*

User initiates creation of a new task (short or long).

User enters **title** of the task.

User enters **detailed description** of the task, if needed.

User selects **foregoing** task(s), which the task depends on.

User sets the **date and time** when the task must begin relative to the completion of all foregoing tasks, if needed.

User sets the **earliest time** when the task can begin, if needed.

User makes all steps that follow setting of date and time in “New short task” and “New long task” scripts.

## 2.6 New subtask for a complex task

User initiates creation of a new subtask (short or long) for some particular task.

User enters **title** of the task.

User enters **detailed description** of the task, if needed.

If the subtask depends on other task(s) or subtask(s), user selects those **foregoing** task(s) and subtask(s).

If the subtask does not depend on other tasks or subtasks, user sets the **date and time** when the subtask must start.

If the subtask depends on other task(s) or subtask(s), user sets the **date and time** when the subtask must start relative to the completion of all foregoing tasks and subtasks.

If the subtask depends on other task(s) or subtask(s), user sets the **earliest time** when the subtask can start, if needed.

User makes all steps that follow setting of date and time in “New short task” and “New long task” scripts.

## 2.7 Repeating task

User initiates creation of a new task (short or long).

User enters **title** of the task.

User enters **detailed description** of the task, if needed.

User sets **time, days and weeks** when the task must be done (for short task) or when the task begins (for long task).

User makes all steps that follow setting of date and time in “New short task” and “New long task” scripts.

## 2.8 Long task split into parts

User initiates creation of a new task (short or long).

User enters **title** of the task.

User enters **detailed description** of the task, if needed.

User sets the **overall time (duration)** required to complete the task – exactly or within some limits.

If needed, user sets **time interval**, within which the **entire** task must be done.

For each part of the task, user sets **start time** and duration.

User makes all steps that follow setting of duration in “New long task” script.

## 3 Task collisions

### 3.1 No time for the new long task

User makes all steps of "New long task" script before saving the task.

User has set not exact time for the beginning of the task, but some time interval instead.

User tries to save the task.

System detects that there is no enough free time in the time interval set for the new task. I.e. the time interval is already filled with other tasks.

System suggests user to save the task anyway, to set another time interval for the task, or to move other tasks filling the time interval.

If user selects to save the task anyway, system shows additional warnings depending on relative importance of the new task and other tasks in the time interval.

### 3.2 Short task colliding another short task

User makes all steps of "New short task" script before saving the task.

User tries to save the task.

System detects that the task falls into time already allocated for another short task.

If importance of the two tasks is different, system sets more important task prior to less important one.

If the two tasks have equal importance, system suggests user either to select which of the two tasks will be activated earlier or to set another available time for one of the tasks.

### 3.3 Short task overlapping long task

User makes all steps of "New short task" script before saving the task.

User tries to save the task.

System detects that the task falls into time already allocated for a long task, which is undesirable or impossible to interrupt.

System suggests user to save the task anyway, to select for the short task another available time, or to move the long task to another available time.

If user selects to save the task anyway, system shows additional warnings depending on relative importance of the two tasks and permission to interrupt the long task.

### 3.4 Long task overlapping long task(s)

User makes all steps of "New long task" script before saving the task.

User tries to save the task.

System detects that the time allocated for the task overlaps time already allocated for another long task.

System suggests user to save the task anyway, to select for the new task another available time, to split the new task into parts, or to move (completely or partially) the long task to another available time.

If user selects to save the task anyway, system shows additional warnings depending on relative importance of the two tasks.

### 3.5 Long task overlapping short task(s)

User makes all steps of "New long task" script before saving the task.

User selected that the new task is undesirable or impossible to interrupt.

User tries to save the task.

System detects that the time allocated for the new task overlaps short task.

System suggests user to save the task anyway, to select another available time for the new task, or to move the short task to another available time.

If user selects to save the task anyway, system shows additional warnings depending on relative importance of the two tasks and permission to interrupt the long task.

## 4 Organizing tasks

## 4.1 Scheduling tasks for the nearest day

User selects a day, which she wants to schedule tasks for.

User selects task **categories** that she wants to schedule.

User looks at tasks already scheduled for the selected day.

System highlights tasks without exact beginning time and with time interval falling (entirely or partially) into the given day.

User can set filter for tasks to be highlighted by importance, duration and deadline.

For tasks without exact beginning time user either sets exact beginning time, or moves a task to some later time, or leaves task as is for later revision.

## 4.2 Scheduling tasks for the nearest week or month

User makes all steps of "Scheduling tasks for the nearest day", but selects required week or month instead of day.

## 4.3 Identifying task overlaps

User selects **time period**, for which she wants to identify task overlaps.

User selects **importance** of the tasks, for which she wants to identify overlaps.

System shows all tasks of selected importance within selected time period and highlights task overlaps.

User **moves** overlapping tasks to eliminate overlaps.

## 4.4 Splitting a long task

User selects a task

User sets the end time for the first part of the task.

User sets date and time for the start of the second part.

## 4.5 New part of a long task

User selects an existing long task, for which time was not allocated completely.

User sets date and time when the part of the task must start.

User sets duration of the part.

# 5 Reminder

## 5.1 Accepted task

When start time a task comes, system alerts user with popup dialog, sound and vibration.

System suggests user selection – either to confirm the task, or to postpone it.

If user is ready to start the task at previously set time, she confirms it.

## 5.2 Postponed task

When start time a task comes, system alerts user with popup dialog, sound and vibration.

System suggests user selection – either to confirm the task, or to postpone it.

If user is not ready to start the task at previously set time, she selects to postpone it.

User sets new start time for the task.

## 5.3 No response from user

When start time a task comes, system alerts user with popup dialog, sound and vibration.

If user does not respond to reminder, system repeats sound and vibration alerts according to alert settings.

# 6 Performing a task

## 6.1 Starting on reminder

System alerts user that a task should be started.

User accepts to start the task.

User performs the task.

System shows the task as **currently performed**.

## 6.2 Starting without reminder

User looks at her scheduled tasks.

User can set filter for tasks to be highlighted by importance, duration and deadline.

User selects a task.

User performs the task.

System shows the task as **currently performed**.

## 6.3 Pausing

User pauses performing the task, when she needs to switch to something else for a short time.

System shows the task as **paused**.

User can either resume the paused task, or stop it.

## 6.4 Stopping

User stops performing the task, if she needs to switch to another task.

System asks user to set the time, when the task will be continued.

If user does not set new time for the task, system marks the task as **interrupted**.

## 6.5 Completing

User completes the task.

System marks the task as **completed**.

# 7 Setting alarm clock

## 7.1 Default alarm settings

For every day of week user settime of the alarm or sets "no alarm".

User setsmelody for alarm.

User setsloudness of alarm.

User setsduration of alarm.

User setsnumber of repetitions.

User setspause between repetitions.

If needed, user setsapplication that will be launched instead or in addition to playing melody.

## 7.2 Changing alarm settings for particular day

User selects one or several days, for which she wants to change alarm settings.

User changes default time of alarm, if needed.

User changes default loudness of alarm, if needed.

User changes default duration of alarm, if needed.

User changes default number of repetitions, if needed.

User changes default pause between repetitions, if needed.

User changes default settings for launching application on alarm, if needed.

# 8 Alarm clock activation

## 8.1 Usual alarm

Alarm clock is activated on time set by user.

System starts playing melody.

User presses any button of CellCo or any key on PC.

Melody stops playing.

## **8.2 Launching application**

Alarm clock is activated on time set by user.

System launches application set by user.

## **9 Synchronization**

### **9.1 PC with PC**

User makes some changes in TimeSafe on PC #1.

User leaves PC #1 turned on.

User runs TimeSafe on PC #2, which has Internet connection with PC#1.

User starts synchronization.

All changes made on PC #1 are transferred to PC #2.

### **9.2 CellCo with PC automatic onconnection**

CellCo is connected to PC

User makes some changes in TimeSafe on either CellCo or PC.

System automatically starts synchronization.

All changes made on one device are transferred to another device.

### **9.3 CellCo with PC automatic without connection**

CellCo is not connected to PC, but is near the PC

User makes some changes in TimeSafe on either CellCo or PC.

System automatically starts synchronization thru Bluetooth.

All changes made on one device are transferred to another device.

### **9.4 CellCo with PC manual**

User works with TimeSafe on her CellCo at home.

User has forgotten her CellCo at home.

From PC at office user initiates synchronization thru Internet with CellCo.

User works with TimeSafe on PC at office.

User leaves PC turned on and goes home.

At home user initiates synchronization thru GPRS connection.

All changes made on PC are transferred to CellCo.