

TABLE OF CONTENTS

History	2
Introducing data	3
Report edition	3
XML import	5
Viewing data	8
User evaluation	8
Result extraction	13
Other user tasks	19
Password change	19
Calendar: edit and copy reports	19
Apendix: how extra hours are calculated?	21

HISTORY

Sep 2001

Igalia S.L. is created and soon appears the need for a task tracking system. An XML DTD and a set of processing scripts are developed and the offline XML reports are used by all the company employees.

Dec 2003 - PhpReport 1.0

Some company employees think that editing XML reports by hand is a hard and uncomfortable task and suggest the idea of creating a web reporting application. PhpReport was born.

Year 2004 - PhpReport 1.1 & 1.2

The time goes by and company staff uses PhpReport daily. The application helps company managers to do project cost and competitive power analysis. Some minor bugs are fixed and some new features, like quick statistics or label editing, are implemented. The desire of publishing PhpReport appears

Jun 2005 - Phpreport 1.3

PhpReport is licensed under GPL and prepared to be published: the app is debianized, SQL auth backend and user administration is implemented and project publishing infrastructure is set up.

Sept 2007 – Phpreport 1.5

Documentation is provided and package debian is improvement in order to use dialog into install process.

INTRODUCING DATA

In this section you will learn how you can introduce data.

- [Report edition](#)
- [XML import](#)

REPORT EDITION

First time you go into phpreport any day this prompt is showed:


 igalia PhpReport

Tasks of the day 23/10/2007

REPORT EDITION

Hour count
Today: 0h (00:00) Week: 0h (00:00)
Sections
<ul style="list-style-type: none">- Report edition- XML import- Result extraction- User evaluation- Password change- Exit
Calendar
23 of October of 2007
M T W Th F S S
1 2 3 4 5 6 7
8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31 1 2 3 4
23/10/2007 Go
Today
Copy reports
23/10/2007
To day
From yesterday

And once you had pulse “new task” button, some like this is printed. You can view that some fields are required:


igalia

PhpReport

REPORT EDITION

Errors exist that must be corrected

Tasks of the day 23/10/2007

Start hour

End hour

Task type

Customer

Project

Task type for this project

Story

Telework
☐

Description:

Hour count

Today: 0h (00:00)
Week: 0h (00:00)

Sections

- Report edition
- XML import
- Result extraction
- User evaluation
- Password change
- Exit

Calendar

23 of October of 2007

M	T	W	Th	F	Sa	Su
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

23/10/2007

Today


Copy reports

23/10/2007

To day

From yesterday

and when finish one task, you can add other:


igalia

PhpReport

REPORT EDITION

Tasks of the day 3/10/2007

Start hour

End hour

Task type

Customer

Project

Task type for this project

Story

Telework
☐

Description:

Hour count

Today: 5h (05:00)
Week: 25h (25:00)

Sections

- Report edition
- XML import
- Result extraction
- User evaluation
- Password change
- Exit

Calendar

3 of October of 2007

M	T	W	Th	F	Sa	Su
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

3/10/2007

Today


Copy reports

3/10/2007

To day

From yesterday

You must save changes (“save changes” button) or you could lost all introduced info in a day if you don’t save the last input.

 **igalia** PhpReport

REPORT EDITION

The changes have been saved correctly

Tasks of the day 9/10/2007

Start hour:
 End hour:

Current hour

Task type:
 Customer:
 Project:
 Task type for this project:
 Story:
 Telework: ☐

Description:

Show all data

Delete task

Start hour:
 End hour:

Current hour

Task type:
 Customer:
 Project:
 Task type for this project:
 Story:
 Telework: ☐

Description:

Show all data

Delete task

New task

Save changes

Cancel

Hour count

Today: 5.5h (05:30)
 Week: 20.5h (20:30)

Sections

- Report edition
 - XML import
 - Result extraction
 - User evaluation
 - Password change
 - Exit

Calendar

9 of October of 2007

M	T	W	Th	F	Sa	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

9/10/2007
Go

Today

Copy reports

9/10/2007

To day

From yesterday

In this point, two fields must be explained: “Task type” VS “Task type for this project”. “Task type” field (type) is a long task into company (by example: coordination, sales, administration, holidays, ...). In other hand, “task type project” field (ttype) is a shortest task from one project (by example: analysis, design, community, implementation, documentation, ...).

Good practices: you can input the data each quarter of an hour. This way, the results are more exactly and they aren’t as perodic numbers... for example 10,333333333333h VS 10,25h.

XML IMPORT

This option allows load a xml file with tasks. There two ways to use xml import:

- load individual xml file
- load a tar.gz file (with several xml file)

In order to load a xml file you must format with a correct title. Format is: “*weeklyreport-user-firstdayintoweek-lastdayintoweek.xml*”, where:

- *user*: user name whom load this tasks
- *firstdayintoweek*: first day into this week (format YYYY-MM-DD), for instance, 2007-11-22. It always must be monday.
- *lastdayintoweek*: last day into week (format YYYY-MM-DD), for instance, 2007-11-28. It always must be sunday.

Process is divided into two phases. Once upload file, it is untar (if tar.gz is provided) and do some checks about formatted title. Next, XML is transformed to PHP file (a XSL rules set do it), and then it executed some queries wich introduce data into database.

Collect reports in XML format	
Select an XML report or a set of them belonging the same user and compressed in tgz format, and press Send when ready. Remember to use always the updated rules.dtd .	
Local file	<input type="text"/> <input type="button" value="Browse..."/>
<input type="button" value="Send"/>	

Sections
<ul style="list-style-type: none">- Report edition- XML import- Result extraction- User evaluation- Password change- Exit
<ul style="list-style-type: none">- Report blocking- Labels- Customer management- Project management- User management- Calendar management- Project evaluation- User evaluation- Management indexes

XML Reports		
Content uploaded to the server. Edit only incorrect files (incorrect name format, incorrect date, etc...).		
Weekly report name	Size	Filter
weeklyreport-admin-2006-01-09-2006-01-15.xml	1526	OK
<div>Process</div> <p> FI : Incorrect date. The report doesn't correspond to a week. IN : Incorrect. File format isn't recognized for that file name. UD : User unknown. The user doesn't exist. NP : You aren't the owner of the weekly report. OK : Report ready to be validated, verified and processed. </p> <div>...</div> <div>Rename</div>		
1 correct filenames and 0 incorrect.		

Sections
<ul style="list-style-type: none"> - Report edition - XML import - Result extraction - User evaluation - Password change - Exit
<ul style="list-style-type: none"> - Report blocking - Labels - Customer management - Project management - User management - Calendar management - Project evaluation - User evaluation - Management indexes

XML reports			
Process XML reports. You can edit and revalidate the reports as many times as needed in order to be finally inserted in the DB			
Weekly report	Valid	Coherent	Saved
weeklyreport-admin-2006-01-09-2006-01-15.xml	OK	OK	OK
All valid reports have been correctly saved			

Sections
<ul style="list-style-type: none"> - Report edition - XML import - Result extraction - User evaluation - Password change - Exit
<ul style="list-style-type: none"> - Report blocking - Labels - Customer management - Project management - User management - Calendar management - Project evaluation - User evaluation - Management indexes

<	Calendar						>
11 of October of 2007							
M	T	W	Th	F	Sa	Su	
1	2	3	4	5	6	7	
8	9	10	11	12	13	14	
15	16	17	18	19	20	21	
22	23	24	25	26	27	28	
29	30	31	1	2	3	4	
11/10/2007					Go		
Today							

Copy reports
11/10/2007
To day
From yesterday

VIEWING DATA

It this section, you will learn what reports you can view.

- [User evaluation](#)
- [Result extraction](#)

USER EVALUATION

REPORTS	BRIEFLY DESCRIPTION
R1	Project / person
R2	Hourly personal (briefly)
R3	Hourly personal (explained)
R4	Person / task type
R5	Person / task type (by project) – Project details
R6	Project / task type (by user) – User details

From this menu, you can view six distinct reports as it's indicated at table above. From “sheets” combo you can select from R1 to R4 reports. R5 and R6 reports are showed when you pulse a project or person name from any others reports.

At first, you must know that dates are always required field. Now, it's explained what each report do:

R1. **Project / person** [\(reports index\)](#)

Report shows worker hours by project over time slot. Rows print all projects which had worked hours, columns show all users.

Sheets:

Start date:

End date:

Name	User				Total result	Percent
	admin	amaneiro	andres	nes		
adm_igalia	7.75				7.75	3.01
debian_course	14.00			45.00	59.00	22.91
phpreport	57.25	45.50	80.00		182.75	70.97
Total result	87.00	45.50	80.00	45.00	257.50	96.89
Percent	33.79	17.67	31.07	17.48	100.00	

Sections
<ul style="list-style-type: none"> - Report edition - XML import - Result extraction - User evaluation - Password change - Exit

Calendar						
9 of October of 2007						
M	T	W	Th	F	S	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4
9/10/2007						Go
Today						

Copy reports
9/10/2007
To day
From yesterday

R2. Hourly personal (briefly) [\(reports index\)](#)

Report prints extra hours by user from worker starts work into company to date indicates into “end date” field. Rows only show staff users, columns show extra hours. You must view special process to calculate extra hours at appendix.

Sheets:

End date:

User	total_extra_hours	Percent
admin	-3857.00	0.00
amaneiro	-1266.50	0.00
andres	-264.00	0.00
nes	-219.00	0.00
Total result	-5606.50	

Sections
<ul style="list-style-type: none"> - Report edition - XML import - Result extraction - User evaluation - Password change - Exit

Calendar
4 of October of 2007
M T W Th F St S
1 2 3 4 5 6 7
8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31 1 2 3 4
4/10/2007 <input type="button" value="Go"/>
Today

Copy reports
4/10/2007
To day
From yesterday

R3. Hourly personal (explained) [\(reports index\)](#)

Report shows extra hours by user over time slot. Rows only show staff users, columns show total hours, workable hours, extra hours for this slot time, and total extra_hours. You must view special process to calculate extra hours at apendix.

Sheets:

Start date:

End date:

User	total_hours	workable_hours	extra_hours	total_extra_hours	Percent
admin	87.00		87.00	-3857.00	0.00
amaneiro	45.50	176.00	-130.50	-1266.50	0.00
andres	80.00	344.00	-264.00	-264.00	0.00
nes	45.00	264.00	-219.00	-219.00	0.00
Total result	257.50	784.00	-526.50	-5606.50	


Sections
<ul style="list-style-type: none"> - Report edition - XML import - Result extraction - User evaluation - Password change - Exit

Calendar
4 of October of 2007
M T W Th F St S
1 2 3 4 5 6 7
8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31 1 2 3 4
4/10/2007 <input type="button" value="Go"/>
Today

Copy reports
4/10/2007
To day
From yesterday

R4. Person / task type [\(reports index\)](#)

Report prints users and their company task types over time slot. Rows show all users, columns show all distinct company task types.



Sheets: Person / task type

Start date:

End date:

User	Type				
	devt	form	adm	Total result	Percent
admin	57.25	22.00	7.75	87.00	33.79
amaneiro	25.50	20.00		45.50	17.67
andres	80.00			80.00	31.07
nes	3.00	42.00		45.00	17.48
Total result	165.75	84.00	7.75	257.50	100.00
Percent	64.37	32.62	3.01	100.00	

Sections

- Report edition
- XML import
- **Result extraction**
- User evaluation
- Password change
- Exit

Calendar

4 of October of 2007

M	T	W	Th	F	Sa	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

4/10/2007

Today

Copy reports

4/10/2007

R5. Person / task type (by project) - Project details [\(reports index\)](#)

Report shows project by user and company task type over time slot. This report prints two frames. First frame is a summary about project info (name, description, worked hours, cost per hour, ...). Second frame shows all users who had participated (rows), and all company task types for each user (columns). It also prints statistic graphics.

Project details			
Data			
Name: phpreport		Invoice: 2400.00	
Description: phpreport tool	Estimated cost per hour: 30.00		
Activation: Yes	Estimated hours: 80.00		
Type	Hours		
	Worked hours	Desviation	
		Total	Percent
all	190.75	110.75	138.44
pex	0.00	-80.00	-100.00
pex+pin	0.00	-80.00	-100.00
Type	Invoicing (cost per hour)		
	Actual/current invoicing	Desviation	
		Total	Percent
all	12.58	-17.42	-58.06
pex	0.00	-30.00	-100.00
pex+pin	0.00	-30.00	-100.00

Sections
<ul style="list-style-type: none"> - Report edition - XML import - Result extraction - User evaluation - Password change - Exit

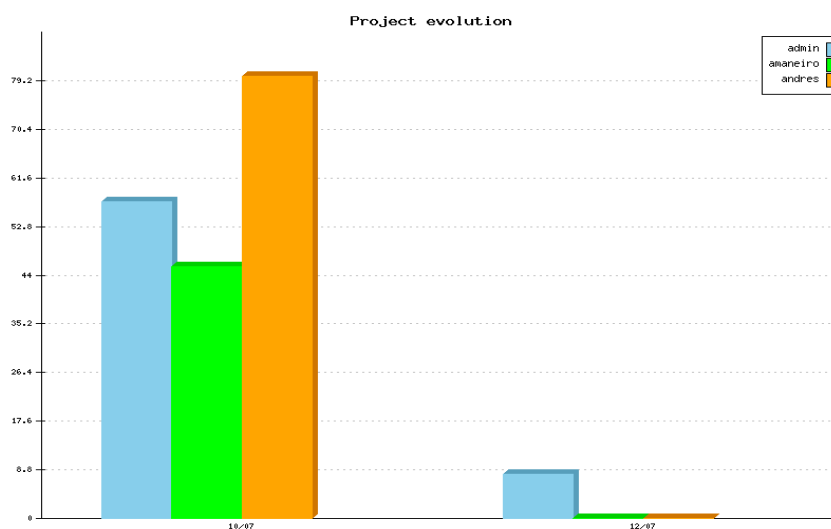
Calendar
4 of October of 2007
M T W Th F S S
1 2 3 4 5 6 7
8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31 1 2 3 4
4/10/2007 Go
Today

Copy reports
4/10/2007
To day
From yesterday

Period to check
Start date: 01/10/2007
End date: 29/12/2007
Set defaults View

User	Type				Total result	Percent
	devt	form	adm			
admin	65.25				65.25	34.21
amaneiro	25.50	20.00			45.50	23.85
andres	80.00				80.00	41.94
Total result	170.75	20.00	0.00		190.75	100.00
Percent	89.52	10.48	0.00		100.00	

Select chart type
bars



R6. Project / task type (by user) – User details [\(reports index\)](#)

Report about user by project and company task type over time slot. Rows show all projects which user had participated, columns show all company task types for each project. It also prints statistic graphics.

USER DETAILS

User details
User Id: admin
Admin: Yes
Staff: Yes

Period to check
Start date: 01/10/2007
End date: 29/12/2007
<input type="button" value="Set defaults"/> <input type="button" value="View"/>

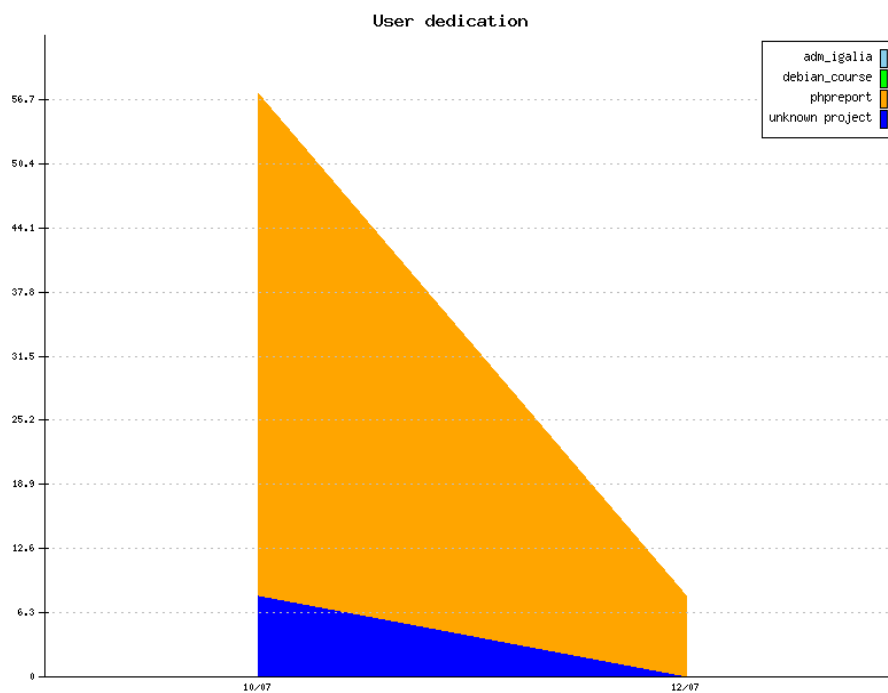
Project	Type				
	devt	form	adm	Total result	Percent
adm_igalia			7.75	7.75	8.16
debian_course		14.00		14.00	14.74
phpreport	65.25			65.25	68.68
unknown project		8.00		8.00	8.42
Total result	65.25	22.00	7.75	95.00	100.00
Percent	68.68	23.16	8.16	100.00	

Sections
<ul style="list-style-type: none"> Report edition XML import Result extraction User evaluation Password change Exit

Calendar
4 of October of 2007
M T W Th F St S
1 2 3 4 5 6 7
8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31 1 2 3 4
4/10/2007 Go
Today

Copy reports
4/10/2007
To day
From yesterday

Select chart type
area



RESULT EXTRACTION

QUERY	DESCRIPTION	USER	ADMIN
Q1	< New Query >	-	X
Q2	Add Compensation	-	X
Q3	Delete Compensation	-	X
Q4	Delete Worker Tasks and Reports (over time slot)	-	X
Q5	Delete all Workers Reports (over time slot)	-	X
Q6	Delete Orphan Reports	-	X
Q7	List all Blocks	-	X
Q8	List all Compensations	-	X
Q9	List all Orphans Reports	-	X
Q10	List all Queries	-	X
Q11	List all Workers Hours (at current week)	X	X
Q12	List all Workers Hours (over time slot)	X	X
Q13	List all Workers Tasks	-	X
Q14	Project Hours	X	X
Q15	Project Hours (over time slot)	X	X
Q16	Project Hours (total VS time slot)	-	X
Q17	Project Stories Duration (by days, hours - no comp)	X	X
Q18	Worker Hours (by date, type - no comp)	X	X
Q19	Worker Hours (by date, project, story, task, type - NULL include)	X	X
Q20	Worker Hours (by date, project, story, type)	X	X
Q21	Worker Hours (by date, project, text, type)	X	X
Q22	Worker Hours (by date, project, task, type)	X	X
Q23	Worker Hours (by date, project, task, type - NULL include)	X	X

Q24	Worker Hours (by date, project, type - no comp)	X	X
Q25	Worker Hours (by project, type)	X	X
Q26	Worker Tasks (by date, project)	X	X
Q27	Worker Tasks (by date, project, type)	X	X

Tricks:

- Format date in database is DD-MM-YYYY
- Hack % is valid in order to fill text fields
- “Task type” VS “Task type for this project”. “Task type” field (type) is a long task into company (by example: coordination, sales, administration, holidays, ...). In other hand, “task type project” field (ttype) is a shortest task from one project (by example: analysis, design, community, implementation, documentation, ...).

Q1. < New Query> ([queries index](#))

View explanation into administrator manual.

Q2. Add compensation ([queries index](#))

This query inserts a new compensation into database. A compensation is a concept created to reduce extra hours in exchange of money. So, when a compensation is granted to a user, the total amount of worked hours is reduced by “num_hours” and the “cantidad_pagada” paid amount is annotated for accounting purposes.

The main part of the queries featured in PhpReport take compensations into account, therefore modifying the total number of hours computed and shown. On the other hand, a few queries don't take compensations into account, and that fact should be bore in mind by the user.

The compensation applies in a limited time range only. So, queries spreading into that time range should take the compensation into account, and queries not spreading into the range shouldn't. That is: if you ask for the worked hours before the compensation range, the hours are taken into account. If you ask for them inside or after the range, the total hours are reduced by the compensation hour amount.

Q3. Delete compensation ([queries index](#))

Delete compensation in database from user “uid_trabajador” over time slot (init date – end date). Delete will be successful only if time slot fit in with compensation.

Q4. Delete Worker Tasks and Reports (over time slot) ([queries index](#))

Delete all tasks and reports from user “uid_trabajador” over time slot (init date – end date). It's necessary to introduce “ok” key at “confirmation” field. It's done to prevent delete reports by chance.

Q5. Delete report from all workers in one intervalo ([queries index](#))

Delete all reports from all users over time slot (init date – end date). It's necessary to introduce “ok” key at “confirmation” field. It's done to prevent delete reports by chance.

Q6. Delete orphan reports ([queries index](#))

Delete reports which don't have to link some task.

Q7. List all blocks ([queries index](#))

An administrator role can blocks the edition report from user over time slot. The users can't to edit the reports at blocked days. This query list bloks from all users.

Q8. List all compensations ([queries index](#))

Compensation is a mechanism wich make posible change worked hours by money, holidays, etc. This extra hours will don't count when you generate “total worked hours recount”. It will does seem that these hours never had exist, because they are compensated.

This query list all compensations.

Q9. List all orphan reports ([queries index](#))

Orphan reports haven't linked some task. This query list all orphans reports.

Q10. List all queries ([queries index](#))

List all queries.

Q11. List all Worker Hours (current week) ([queries index](#))

This query calculates the worked hours from all users at current week. The compensated hours at current week are descouted.

Q12. List all Worker Hours (over intervale) ([queries index](#))

Calculate the worked hours from all users over time slot (init date – end date). The compensated hours over this slot time are discounted.

Q13. List all Worker Task ([queries index](#))

List all worker tasks order by user and date.

Q14. Project Hours ([queries index](#))

Show worked hours from a project (or all projects if you use % hack).

Q15. Project Hours (over time slot) ([queries index](#))

Show worked hours from a project over time slot.

Q16. Project Hours (Total VS Time slot) ([queries index](#))

Compare total hours from a project and hours from a project over slot time. It's only showed projects which they have tasks at slot time indicated.

Q17. Project Stories Duration (by days, hours)(NO COMP) ([queries index](#))

Show days and hours from each project story. This query don't take into account compensated hours.

Originally, compensated hours were taken into account, but it caused wrong conclusions about real worked hours into a project. For this, now don't take in account compensated hours.

Q18. Worker Hours (by type and date)(NO COMP) ([queries index](#))

Calcule worked hours from a user by a company task (“type”) over time slot (init date – end date). This query don't take into account compensated hours.

Q19. Worker Hours (by date, project, story, task, type) (NULL include) ([queries index](#))

Calcule worked hours from a user by a company task (“type”), project task (“task”), project and story project over time slot (init date – end date). It's useful query in order to view projects hours by use case.

The NULL clause indicates what the fields “task” and “story” return NULL values too. For instance, if you filter the query where “task” value equals to "analysis" and “story” value equals “test-c1”, they returns registers which “task” value is "analysis" or null and story value is “test-c1” or null.

Q20. Worker Hours (by date, project, story, type) ([queries index](#))

Calculate worker hours over time slot (init date – end date) by company task (“type”), project and project story.

Q21. Worker Hours (by date, project, text, type) ([queries index](#))

Calculate worker hours over time slot (“fecha fin”-“fecha inicio”) by company task (“type”), project and project task text (task with same text are grouped).

Q22. Worker Hours (by date, project, task, type) ([queries index](#))

Calculate worker hours over time slot (init date – end date) by company task (“type”), project task (“task”) and project.

Q23. Worker Hours (by date, project, task, type)(NULL include, NOT COMP) ([queries index](#))

Calculate worker hours over time slot (init date – end date) by company task (“type”), project task (“task”) and project. This query don't take into account compensated hours.

The queries Q22 and Q23 seem the same but they aren't. "Date", "project" and "type" fields are required, but not "task" fields. That's what make them different:

- *Returning registers.* If you execute Q22 query and there is a register in database wich “task” value is null, the execution result don't return this register. Even though, the Q23 execute result returns the null task field too.
- *Filtering registers.* If you filter the Q22 query where “task” value equals to "analisis", it only returns registers which “task” value is "analisis". Even though, the Q23 query returns register which “task” value is "analisis" or null.

Q24. Worker Hours (by date, project, type)(NO COMP) ([queries index](#))

Calculate worker hours over time slot (init date – end date) by company task (“type”) and project.

Q25. Worker Hours (project, type) ([queries index](#))

Calculate worker hours over time slot (init date – end date) by project.

Q26. Worker Tasks (by date, project) ([queries index](#))

List worker tasks over time slot (init date – end date) by project.

Q27. Worker Tasks (by date, project, type) ([queries index](#))

List worker tasks over time slot (init date – end date) by project and company task (“type”).

OTHER USER TASKS

- Password change
- Calendar: edit and copy reports

PASSWORD CHANGE

It's easy knows that it goes. Into database is stored hash password (make with md5 postgres function, view technical manual).



PASSWORD CHANGE

Input the new password	
Current password	<input type="text"/>
New password	<input type="text"/>
Confirm password	<input type="text"/>
<input type="button" value="Change"/>	

Sections
<ul style="list-style-type: none">- Report edition- XML import- Result extraction- User evaluation- Password change- Exit

<	Calendar						>
4 of October of 2007							
M	T	W	Th	F	Sa	S	
1	2	3	4	5	6	7	
8	9	10	11	12	13	14	
15	16	17	18	19	20	21	
22	23	24	25	26	27	28	
29	30	31	1	2	3	4	
4/10/2007					<input type="button" value="Go"/>		
Today							

Copy reports
4/10/2007
To day
From yesterday

CALENDAR: EDIT AND COPY REPORTS

Within calendar, you can edit and copy reports day to day. In order to edit a specific report, you must go to this day and then you can edit the daily tasks. If you can't edit these task, it's possible that an administrator had blocked the task edition. You must get in contact with your administrator.

Two modes exist to copy reports:

- “Day-to-day” mode

First step: go to day from you want copy. Second step: at “Copy reports” menu you must select day to copy and then pulse “To day” menu.

- “From yesterday” mode

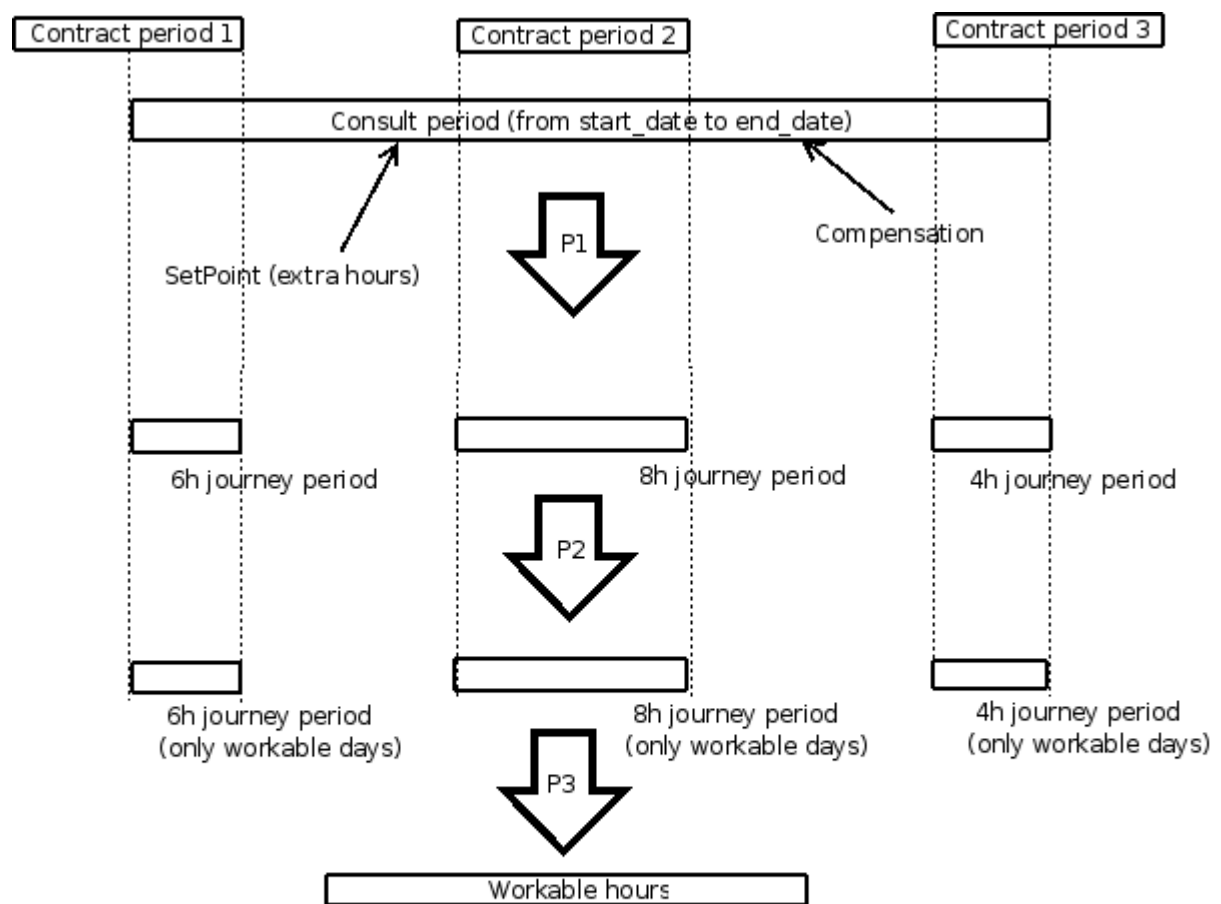
You must go on day in which you must copy tasks. Then, pulse button “From yesterday” and yesterday report will be copy at day what you had selected.

<		Calendar					>	
4 of October of 2007								
M	T	W	Th	F	St	S		
1	2	3	4	5	6	7		
8	9	10	11	12	13	14		
15	16	17	18	19	20	21		
22	23	24	25	26	27	28		
29	30	31	1	2	3	4		
4/10/2007						Go		
Today								

Copy reports
4/10/2007
To day
From yesterday

APENDIX: HOW EXTRA HOURS ARE CALCULATED?

- $\text{extra_hours} = \text{worked_hours} - \text{workable_hours}$
- $\text{worked_hours} = \text{sum}(\text{"all time tasks"})$
- workable_hours is a very little complex calculation. Figure below indicates how it goes:



P1: clip periods out of range

P2: leave out weekend and festives days

P3: add all them

extra_hours can be positive or negative value. If they are positive, it means that worker has worked more hours than she/he must have at this slot time. If they are negative, it means that worker has worked less hours than she/he must have at this slot time.

In other hand, there is two special thing about extra_hours calculation: setpoint and compensation.

Setpoint. Into a table in database (table extra_hours, view technical manual) is stored a value (only to certain users) which indicates number of extra_hours for a certain date. When extra_hours is calculated, this value is taken as initial reference to calculation. For instance, if user admin has this value as 1789h at 13-april-1934, this hours are init extra_hours and hours dates before will be ignored (they could be different than zero). This value have to be set from a client database for your database server, you can't do this from phpreport application.

Compensation. This is a mechanism wich make posible change worked hours by money, holidays, etc. This extra hours will don't count when you generate "total worked hours recount". It will does seem that these hours never had exist, because they are compensated.