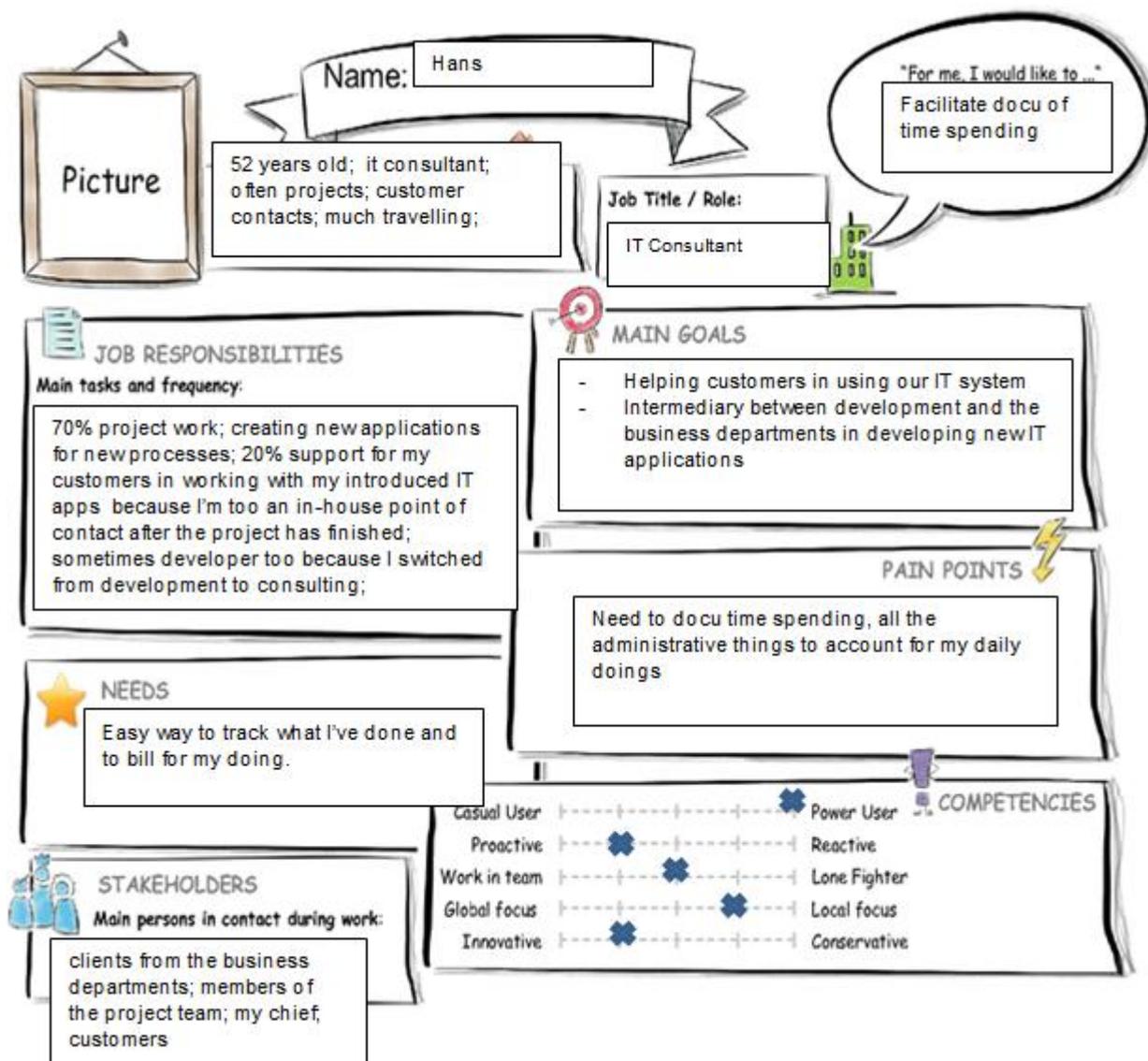


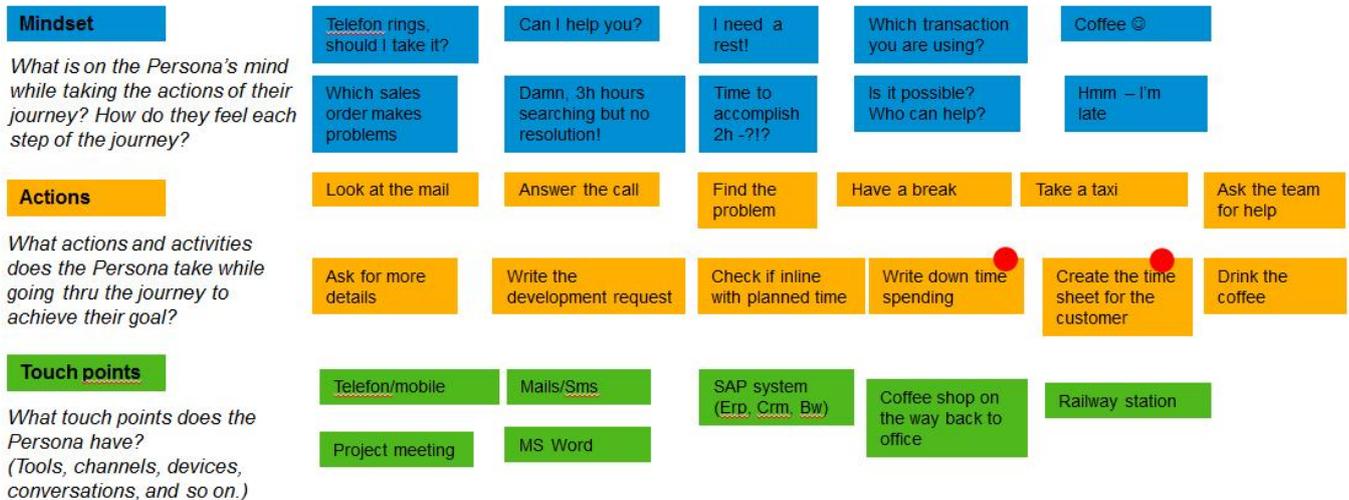
1 Story

I am a (very expensive :-)) IT consultant and have to document the time (almost too much) I need for a specific task. There is a lot of project work but also normal office stuff, support of colleagues, travel time, coffee pauses and my chief needs sometimes quick a presentation for his next meeting. In the evening I often ask myself what I have done the whole day and then I believe the most creative time of my day starts: to fill out the timesheet for the day.

2 Persona



3 User Experience Journey - Duration of the Journey: one day



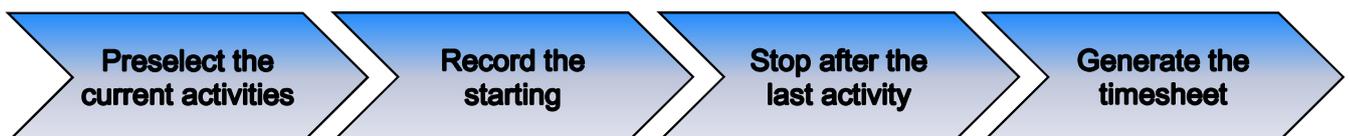
There are a lot of different activities during a day and often something intervenes. Despite of that, the time sheet for the day should reflect the daily doing.

4 Mock Up

The idea for the app:

Use a running clock to record the time spending for an activity. The app allows capturing the start of an activity. One activity ends when the next one starts. It is not necessary to record the end of an activity as long as another one starts.

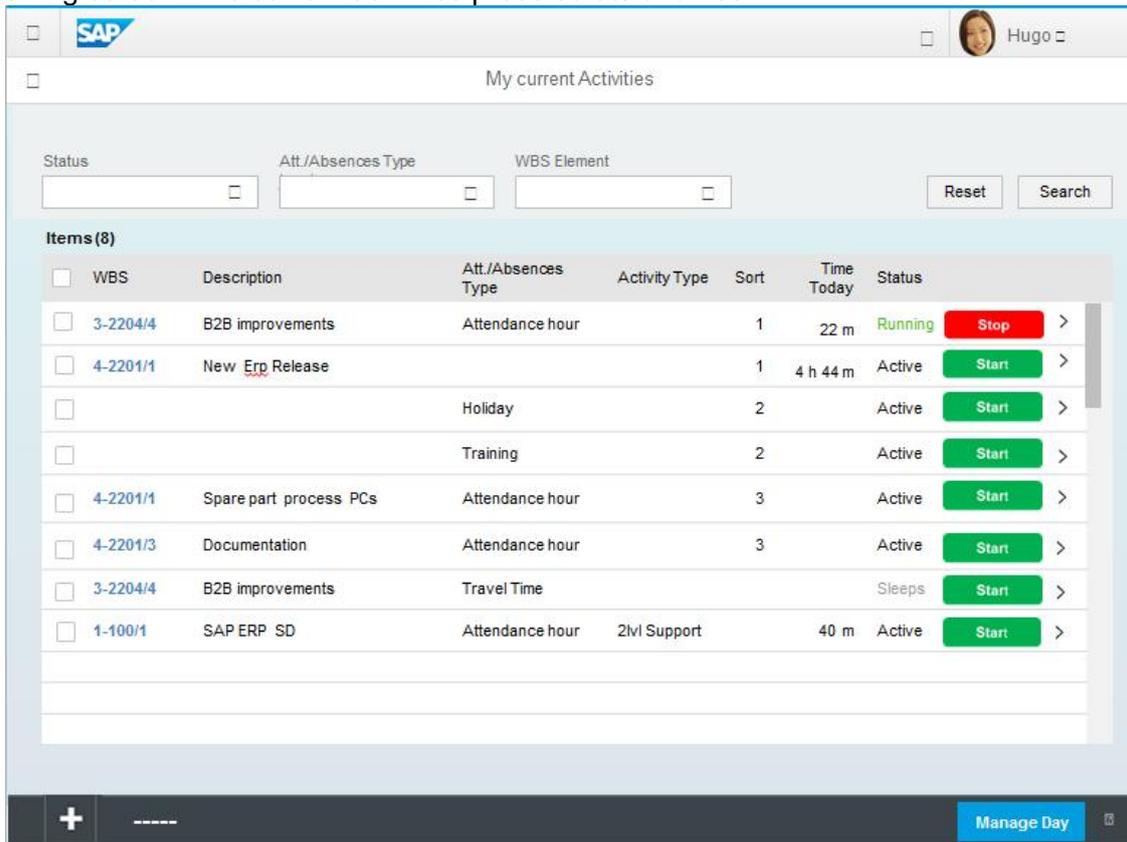
At the end of the day it is possible to check all the time slices recorded during the day. You can correct manually all the data but that should mostly not be necessary. Out of all the recordings it is possible to generate the time sheet data for the day.



Steps supported

The persona has a lot of different tasks to do during the day. To stay on top of the things, at least for the issue time spend for what, the app should have only a small distraction factor during the daily doing: call the app if not already open, start the activity by pressing a certain button almost always on the first screen.

The starting screen - the current activities preselected/favorites



Some features:

- Individual preselection of the current activities for the user
- With the actions “+ (new)” and “- (delete)” the user can add a new activity or delete an existing one from his pool.
- The “> Details” opens the detail screen for an activity with all the data in a form format and the possibility to change everything.
- The data:
 - What: WBS and WBS description, Attend./Absence type, Activity type
 - Sort: To prioritize an activity over another (relevant for sorting).
 - Status: Active, Sleeping, Running (relevant for filtering)
 - Time today: Today totals for the activity – calculated from the individual time records

- Actions “Start/Stop”

“Start” creates a time record for the selected activity/row. All the already created records are listed on the next screen. E.g. the selection of the 2nd row “New Erp Release” in the screen above and action “Start” at 08:00 in the morning have created the 1st row in the following screen with the data “WBS, Att./Abs.Type, Activity Type and Start”. The other fields (description and time) are calculated.

Items (7)								
WBS	Description	Att./Abs. Type	Activity Type	Start	End	Time	Time Sheet	
4-2201/1	New Erp Release	Attendance hour		08:00		22 m		>
1-100/1	SAP ERP SD	Attendance hour	2M Support	08:23		20 m		>

“Stop” adds the end data for a running activity (only possible for running activities).

- The “running” activity is always listed as the first row.
- Action “Manage Day” calls the next screen with all the individual time records already created and the possibility to correct everything.

The 2nd screen - lists all the created time records for the day

WBS	Description	Att./Abs. Type	Activity Type	Start	End	Time	Time Sheet
<input type="checkbox"/> 4-2201/1	New <u>Erp</u> Release	Attendance hour		08:00		22 m	>
<input type="checkbox"/> 1-100/1	SAP ERP SD	Attendance hour	2lvl Support	08:23		20 m	>
<input type="checkbox"/> 4-2201/1	New <u>Erp</u> Release	Attendance hour		08:43		3 h	Yes >
<input type="checkbox"/> 1-100/1	SAP ERP SD	Attendance hour	2lvl Support	11:43	12:02	20 m	Yes >
<input type="checkbox"/> 4-2201/1	New <u>Erp</u> Release	Attendance hour		13:00		1 h 22 m	>
<input type="checkbox"/> 3-2204/4	B2B improvements	Attendance hour		14:23		22 m	>

Some features:

- Lists all the time records created for a certain day until now.
- With the actions “+ (new)” and “– (delete)” the user can add manually a new activity or delete an existing one.
- The “> Details” opens the detail screen for a time record with all the data in a form format and the possibility to change everything. There is too the possibility to add a comment.
- The value “time” is calculated from the app.
- Action “compress”
For the timesheet all the detail start/end times are not relevant. With the action for all the marked rows the program sums up the individual times so that only one row per activity remains.
The compressed records are not written into the database, they are only used temporary for creating the timesheet records. The compression could be revoked.
- Action “Create Timesheet”
For all the marked records (even if compressed) a time sheet record will be created. Transferred records get the status “Timesheet=yes”. Only not transferred records are taken.

By using only the start time it is easier to correct a record if the user forgets to press the start button if an activity changes. For a delay of 10 minutes e.g. it is only necessary to shift the start time a little bit in the past.

Remarks:

- A little chart would be nice to check how much the user has spend on a certain activity (selectable: daily, weekly, monthly)



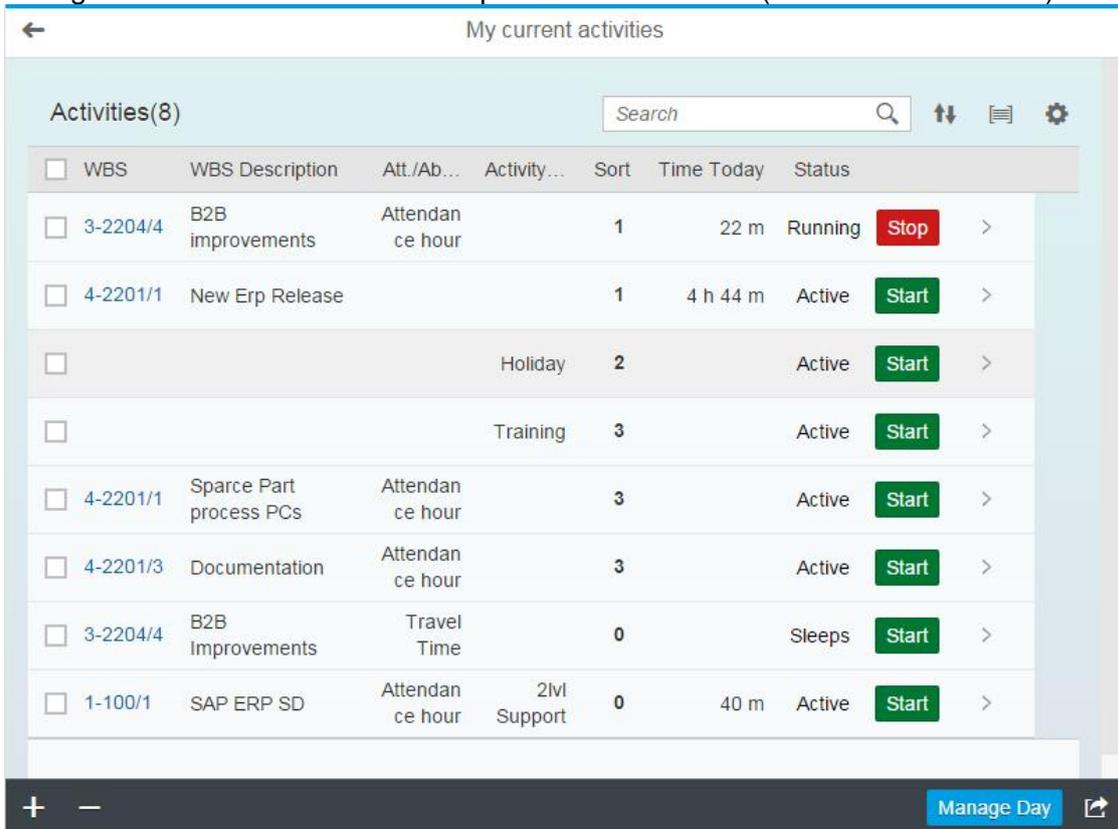
- All the time records are a good database to create the weekly or monthly summary for the customers. For that, surely there are additional data necessary, e.g. if a time recorded is billable. Because the recorded time slices are only loosely coupled to the time sheet, enhancing the data should be no problem.

Some further aspects of the app:

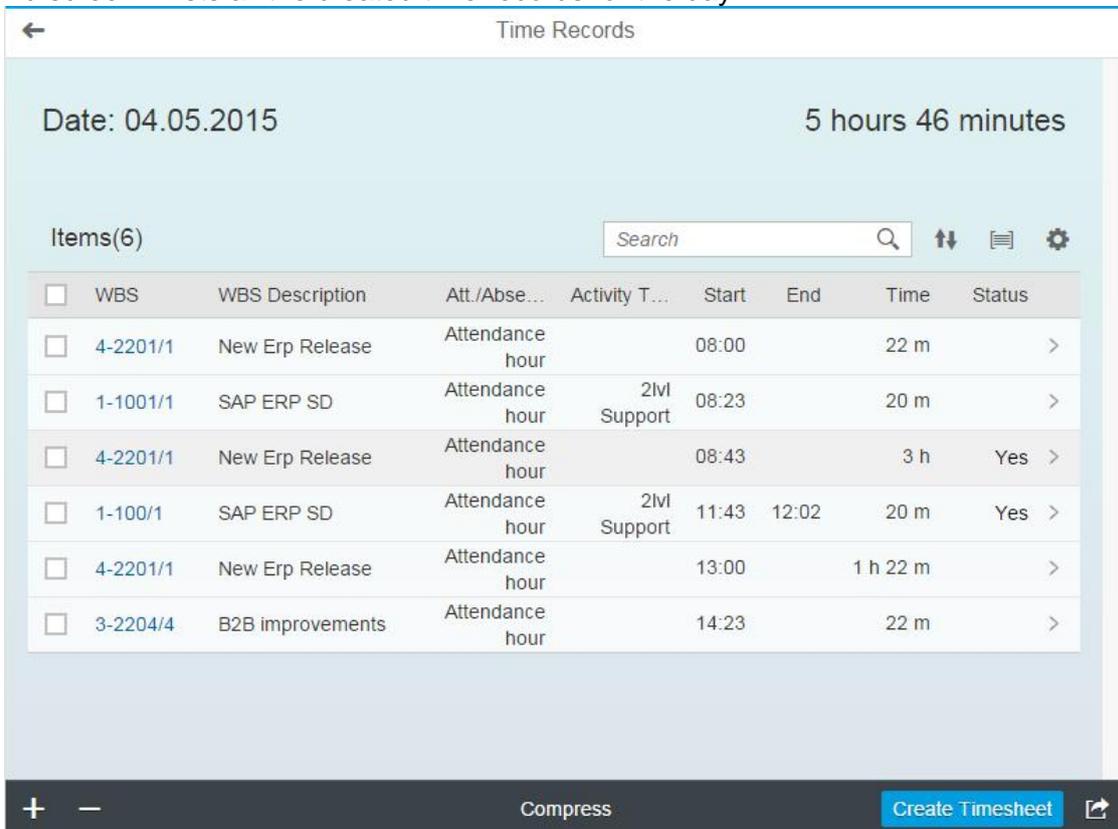
- Not much to do in the main screen, some presets (day, preselected activities).
- A lot is created in the background, but room enough to manipulate created data.
- Loosely coupled to existing processes.
You can use the recording, but you do not have to do it. E.g. you can use the recording only for the project work.
- Without the analytic part only a light app. One table for the preselected activity for the user and one for the daily time slices already recorded.
- Beside of the domain values for the 2 status fields, no further master data necessary.

5 App Prototype

The starting screen - the current activities preselected/favorites (here without filter bar)



The 2nd screen - lists all the created time records for the day



6 Just other nice to have features for our Persona

Transaction code/ App code

I really liked the concept of the transaction codes in an ERP-System and use it a lot. Why not use it in the app world too?

Our persona, the inhouse consultant with support tasks, e.g. gets a lot of inquiries for a special order, often by phone. Almost always he tries to reproduce the problem in his system by calling the same transaction/app and the same screen. Surely there is the concept of the url, when send by the end user, makes it possible to jump to the place where the user is at the moment. But at your example apps it is not always working. And on the phone the end user tells just the number, sending the url means an additional step.

In the app world, the following steps are necessary to call a special order:

- Find the app in the Launchpad
 - by searching for the app by entering part of the app title
 - by using favorites, groups, recently called ...
- Click on the app tile
- Enter the order number in the search field
- Click on one of the entries in the result list

Only necessary is:

- .sdo 36114715

What does it means:

- Call the app which belongs to the transaction/app code SDO (SD=SD-component, O=order), or any other code suitable.
- Pass the number to find the relevant order and already open it if found.

Technically it should be no problem:

- The leading dot entered in the search field tells the system that a transaction/app call follows (the dot because it is easy to reach on every keyboard even on mobiles).
- With the app-Id you already have a technically name for the app.
- The passing of parameters should not be a problem for html apps.
- So you only need the link to a transaction code and
- certainly the handling of the parameters in the app.

Context apps

One of the main ideas of a Fiori app is the simplification, which means focusing on the relevant information of a specific task and strip information which are not used in general.

For our persona, the consultant, that means, that there are more transactions/ apps to supervise. To find a problem often it is necessary to have additional information from processes linked to the problem. E.g. if one of the accounting colleagues asks why an invoice is not triggered, it is often necessary to follow the flow to the beginning of the process.

Apps should be lightweighted, so an app should not be overloaded to fulfill all possible requests when another app is more suitable. So why not make it possible to link other apps with more or other information to an app so that it is possible to jump around to find the appropriate information. Ideally the key fields for the next app are already passed, so that you can jump directly to the relevant business object.

A place for the context apps could be where the configurations settings are.