

We now have federated timesheets: <https://github.com/federatedbookkeeping/timesheets>

### Table of contents

---

- [Roadmap](#)
    - [Preliminary phase](#)
    - [Development phase](#)
    - [Follow-up phase](#)
  - [Wishlist](#)
    - [Open](#)
    - [Pending](#)
    - [Closed](#)
  - [Who](#)
  - [Documentation](#)
  - [Other wiki-based wishes & infos](#)
  - [Collections of ideas](#)
  - [List of decentral Android ToDo-list or Timesheet apps \(clients\)](#)
  - [Related](#)
- 

### Roadmap

#### Preliminary phase

1. When jTrack is not suitable anymore: Research (if any) available PWAs and libs are ready to use (license compatible, dev phase requirements met)
  1. if not: Research if a "PWA framework" is necessary to develop one
    1. if yes: choose one:
      - Ionic?
      - Polymer?
      - ReactJS?
      - VueJS?
      - ...?
    2. else: develop in pure HTML5 and JS/jQuery from scratch (with Bootstrap classes)
  2. Do we need dedicated features integrated/enhanced to achieve everything we need or Profile(s) would do the same job?

#### Development phase

1. If an existing PWA already available to use
  1. add to packagist and integrate with Trackers
  2. else: code one (Fabio) as a proof of concept aka version 0.1 and integrate with Trackers

### Requirements

The solution must consist of two main parts:

1. a Time Tracking app (to measure the time spent on projects)
2. Time Sheet (Time Tracker) to store the measured data

Whether a new solution is needed or not, here are the requirements:

- Time Tracking

- Simple, nice, easy to use interface (see Toggl for inspiration)
  - Task name
  - Project name
    - Client name
  - Start/Stop button (keep focus only on one task at a time)
  - Continue button on previously tracked items
  - All fields editable (details in pop-up? x-editable lib could be used?)
- Must work in any modern browser and desktop (OS independent - native app or at least browser extension needed or just browser window?)
- Must work on iOS and Android (native apps needed or just web app in native browser?)
- Must warn user if screen is idle for specified amount of time (keep running/discard since the idle time)
  - Browsers can display desktop notifications nowadays (see <http://stackoverflow.com/questions/2271156/chrome-desktop-notification-example>)
  - Push notifications possible too?
  - E-mail notifications?
- Sync with Tracker - store data to display the TimeSheet summary/reports
- Time Sheet (a tracker only solution or needs a dedicated feature?)

## Integration

In phase one, develop a PoC PWA Timer (Time Tracker).

Phase two, work with the tracked data and re-use in Reports, Invoices, Payments and related features...

## Packagist

...

## Development

Fabio is going to develop the PWA with luci's supervision.

...

## Profiles?

- If integration can be made using Tiki Profiles and existing features - code the profile(s) in YAML on [profiles.tiki.org](http://profiles.tiki.org)

## Follow-up phase

- Announce on dev list, article and social channels
- Public beta testing
- Write docs

---

**The information below might be out-dated but is kept for review:**

---

jTrack was added to [Tiki9](#) (yay!), now let's progressively make it better and integrate with the rest of Tiki (task tracking, invoices, etc.)

## Use cases

- a company which hires freelancers on projects (and some of that time needs to be re-invoiced to the

end customer)

- manage time sheet for 15-person not-for-profit association.

## Wishlist

### Open

Rating	Subject	Submitted by	Importance	Easy to solve?	Priority	Category	Volunteered to solve	Created	LastModif	Comments
● ★★★	<a href="#">Commit Timesheet data to the linked Tracker doesn't transfer expected duration values, just the text and values from timers</a>	Xavier de Pedro	6	6	36	• Feature request • Support request		2014-10-21	2014-10-21	0
● ★★★	<a href="#">Timesheet CSS should load before other style sheets so its body properties can be overridden</a>	Gary Cunningham-Lee	7	7	49	• Usability		2016-04-14	2016-04-14	4 chibaguy-13 Mar 17

### Pending

Rating	Subject	Submitted by	Importance	Easy to solve?	Priority	Category	Volunteered to solve	Created	LastModif	Comments
--------	---------	--------------	------------	----------------	----------	----------	----------------------	---------	-----------	----------

### Closed

[+]

Who

- Regis Barondeau
- Marc Laporte
- Kimberly Fink
- Michael Pilling
- Xavi (*his boss at work is highly interested in finding some time tracking solution for this work team, out of Tiki if Tiki doesn't provide it*)
- Robert Plummer
- luci

### Documentation

Documentation of the current implementation for [Tiki9](#) is here, as a work in progress:

## Other wiki-based wishes & infos

### Options

1. extend [trackers](#), like [http://profiles.tiki.org/Time\\_Sheet](http://profiles.tiki.org/Time_Sheet)
2. use [Spreadsheet](#)? (for calculation, etc)
3. new standalone features?
4. A combination? (add time in trackers, make calculations in spreadsheet, and then, send to [Payment](#)).

### Feature requests

- How to send invoiced time to [Payment](#)?
- Have a way to have a running total (so hours are added as they are done, but a negative entry is done when an invoice is prepared)
  - For freelancers: thus, the customer knows at any time how many hours are done (and will later be billed)
  - For staff: staff typically get a fixed paycheck for a fixed number of hours. However, the actual number of hours worked will fluctuate. So each paycheck would influence the number of hours in the bank. Vacation, holiday, sick, personal days, etc will also influence. So both employee and organization know how many hours are in bank or owed. This needs to be flexible because rules can change over time and are different from organization to organization.
- Ability to import data from csv, since in some cases, the person logs its time in local on a spreadsheet, and then, data sets can be added (imported) into the time sheet at once.
- Somehow, allow to relate records in the time sheet to projects (from a db of projects stored in a tracker?)

### Mike's thoughts

(based on observations from person and my software consulting on time tracking)

- calendar based time tracking (like google calendar) is very popular and the closest thing to the classic "daytimer" so calendar based (calendar tracker integration?) would be a big wish for this.
- timecards need pending and approved or finalized status. Items are not billable / payable until they have been approved.
- adding a charge from a defined list of products/services to a timeblock would be great.
- timeblocks could be associated with projects and or clients (like tags or categories)
- ical integration for outlook / goocal would be awesome.
- one set of tags/ categories should apply to "status" - a simple workflow of "pending/tentative, active (to be done), done, closed, canceled. Trackers use of pending active closed may be too hardwired or can we expand the number of statuses?

### Robert Plummer's thoughts

I just integrated a working example of "Time Sheet" using jTrack, which is a plugin that is designed to track time locally in your browsers cache. I've used TrackerQueryLib to pull tracker data from the [Time Sheet Profile](#), my next step will be to create a plugin that can be used in a page, and then to extend to be used within [Invoice](#) as a means of creating line items.

cdrwhite: Analysis/Requirements for timesheets

This has been translated from our german intranet page

### Existing solutions

This is an incomplete list of current time sheet packages with web interface and or API. *Kimai* and *mite*

look convincing, and *JTrack* looks good enough for basic use cases, combined with trackers (like in profile `Time_sheet`).

Name	Description	Language	webbased?	License?	Commercial?	Extensible?	Fits my needs?	Convincing?
<a href="#">Kimai</a>	record work time	PHP	Yes (no API)	GPL	No	Yes	90% (no API)	1
<a href="#">JTrack</a>	Jquery Plugin for Time Tracking	js (jq)	yes	MIT	no	? (I guess so)	yes if linked to a Tiki tracker such as the one at the <code>Time_Sheet</code> profile (xavi)	1
<a href="#">Time4U</a>	platform independent, desktop- and web, Central data storage, desktop client offline, Open Source	Java + Tomcat	Yes + API	MIT License	No	Yes	97% (Installation/Hosting)	2
<a href="#">ClockingIT</a>	Time sheet, project management and task management	Ruby on Rails	Yes (no API)	MIT/X11-Lizenz	No	Yes	90% (API can be added)	2
<a href="#">Time-O-Meter</a>	web based, project oriented with minimal effort	PHP	Yes (no API)	GPL 3	No	Yes	85% (No API)	2
<a href="#">openTimetool</a>	openTimetool is a web based time recording for projects	PHP	Yes (API unknown; don't think so)	GPL	No	Yes	80% (No API/customer access?)	3
<a href="#">mite</a>	timesheet with functionality, flexibility & elegance	-	Yes + API	Commercial (p.P. 5,00 € per month)	Yes	No*	99% (time sheet, API, customer access.)	1
<a href="#">Harvest</a>	Business time tracking & invoicing	-	Yes + API	Commercial (12-90\$ per month)	Yes	No*	99% (API, customer access)	2
<a href="#">LogMyTime</a>	time sheet for self employed people, small business	-	Yes + API	Commercial (p.P. 6,95 € per month)	Yes	No*	75% (No customer access)	2
<a href="#">Timesheet.php</a>								99
<a href="#">Timesheet NG</a>								99
<a href="#">Anuko Time Tracker</a>								
<a href="#">Timetrex</a>								
Open apps like OrangeHRM have time sheet modules								

## Remarks:

*Convincing?*: 1=yes .. 3=has potential ... 5=no, 99=unknown

*No\**: A php script on a local server (in tiki) could interface via API with the commercial provider.

Related: [Invoice](#) and [Payment](#)

Collections of ideas

My Requirements (cdr white)

- a database server (MySQL/PostgreSQL/PHP accessible) as backend
- objectoriented PHP-library provides the core functionality
- the library can be accessed via REST-API and a web interface
- possible clients/interfaces:
  - API for external applications
  - Web-Interface
  - iPhone-App
  - Firefox-Plugin
  - Phone-API
  - Hardware terminals with chipcard
- Each project should have several customers, who will receive invoices
- Each customer can have n users (employees/contractors/freelancers) and projects, A customer is automatically the first employee for himself and has a global project
- Each customer can add employees to projects and define internal and external "prices" for the recorded times
- Each project can have different categories, e.g. "Development", "Meeting", "Documentation"
- Each customer can have multiple clients and give them read access to projects (external times/prices), a client will be another customer of the service
- Times must be recorded via one click (log in/out) and by entering start and end times

Torsten & Michiel de Jong at 21st Oct '22 on gitter

- The demonstration of [George Svarovsky on YouTube](#) is a great proof of concept and might be partially good for productive use, as the tool timeld seems to be mainly for commandline use and not very trivial to install for non-techie endusers. Anyway how Tiki and timeld work together is just awesome.
- For many use cases and people it would be important to have an easy to install, simple and comfortable interface on their phone to submit their tasks and hours
- One example of such an app would be TeamWork from teamwork.com with the massive downside, that even if an API between Tiki and Teamwork would be developed (seems to be under way), still a SAAS would be involved. If SAAS would be involved,
- Our FOSS approach is to provide customers and (developers aswell) a maximum of data sovereignty and data security and to respect the GDPR.
- Torsteb and Michiel agreed, that it is necessary to find one or several FOSS mobile apps, ideally for Android and iOS, tha are able to sync with "something". That "something" could be open source self-hostable software, or some open protocol like CalDAV.

Security

The data being processed is relevant for the german BDSG, as these informations are personal informations. Therefore the need for security is high/very high, it must be assured that data manipulation/read will be done by the right people.

Web based communication will be encrypted via https, maybe even with client certificates.

## Distributed

- In a project, there could be a freelancers, a web agency and a client. The timesheet information should be available to all three.
- Should be easy to have a running total of time per project
  - Total done (by team member)
  - Estimated done so far (but not received time sheet or invoice)
  - Total done + estimated done
  - Estimated for the rest of the project (by team member)
  - Total estimated
- From the time sheet, should be easy to make an invoice, and from that invoice, a [Payment](#).

So we need [Distributed data](#)

## Valerie's comments

I've been thinking about this in terms of the smartphone as the most likely input device for time tracking issues, that ultimately need to seamlessly be captured into accounting and tax preparation solutions.

Some people will simply use a calendar and most likely something like Outlook or Google rather than Tiki.

Others will prefer to use a stop watch on their phone that is captured as a time sheet. The truly technical junkies will want to use the GPS to track their mileage.

The problem with time tracking for the individual today is that inevitably different groups relevant to the professional and personal realms settle on different solutions. It is difficult to manage this through a single interface with the result that things that could be simple to handle once in real time, end up getting backlogged.

Therefore, flexibility to let users choose their own solutions for key components (calendars, task lists, time and mileage trackers). So while Tiki in many respects DOES project management, the individual participants in the collaboration should ideally be able to use their choice of PIM-type tools. Support for standards and integration are critical to achieving this goal.

My investigations into this issue have led me to conclude that Toggl is worth looking into. It has a very nice simple interface, an API, and attractive pricing, including an exceptionally generous entry level free option (5 users/unlimited projects).

At this point I am not sure that it is worth integrating Toggle with Tiki (it depends on how much the reporting could be handled by other Tiki features and how valuable integration with Basecamp would be ... I am assuming that iCal and RSS integration isn't so valuable).

## PluginTimeSheet idea

Perhaps a spreadsheet or a tracker but what about a wiki plugin?

## Benefits:

- Less overhead than firing up a tracker
- Still makes total
- Easy to copy/paste to another site
- Can be used in any Tiki project, and people watching the page can receive email notifications
- Eventually could be used [Offline](#)

## Features to be coded

- "Time Worked" would make a total
- "Time Worked" could be used as a variable in a page to add up values

#### PluginTimesheet

```
{TIMESHEET(user=Sam, category=can come from tracker data, categories, etc.)}
Date and time start | Date and time end | Time worked | Description | Category
{TIMESHEET}
```

### TikiFest 2013 Report

1. Timesheet
2. Expenses reports
3. Tasks

#### Employee or contractors

- Keep track
- Time
- What was invoiced today
- Their client

- Date { time entries }
- # Hours (\*1)
- Client
- Project/Dept.
- Task/Description
- Non billed hours
- Start & stop time (\*1)
- Hourly rate?
- Expense report?

-> invoice numbers it relates to -> contractor -> agency  
 -> invoice numbers it relates to -> agency -> to customer

#### Customer

- Dashboard/Report
- # Unbilled hours & detail of timesheet
- # Billed hours
- Printable
- Watch or weekly report

#### Agency

- Report per project with aggregate of all contractors
- Bank of hours vs. invoice



List of decentral Android ToDo-list or Timesheet apps (clients)

An Android app to work in a federated environment **must** have a capability to sync tasks with FOSS Timesheet apps in some way, for example using CalDav or other common techniques. **(more to be added)**

## Tasks

Latest version on F-Droid: 12.7 (120700) submitted at 2022-07-03

<https://f-droid.org/de/packages/org.tasks>

GNU General Public License v3.0 only

<https://tasks.org>

Successor of **Astrid**

Astrid was a popular cross-platform productivity service that was acquired and discontinued in 2013. The source code from Astrid's open source Android app serves as the basis of Tasks.

- Open protocols  
Tasks.org is compatible with Outlook, Thunderbird, Apple Reminders, and more!
- Powerful and flexible  
Filters, tags, lists, infinite-depth subtasks, manual sorting. Tasks works with nearly any productivity system!  
can use Markdown

Tasks.org is compatible with Outlook, Thunderbird, Apple Reminders, and more!

- Customizable and multilingual  
Extensive options for customizing the look and feel of Tasks. Use Tasks in your language
- Multiple sync options  
Synchronize with Tasks.org, Google Tasks, DAVx<sup>5</sup>, CalDAV, EteSync, or DecSync CC
- Private  
Use offline, self-host, or setup EteSync for end-to-end encryption. There are no ads and your data is never shared with anyone

## Pendulums

Version: 1.1.0

<https://github.com/Swing-team>

GNU General Public License v3.0

<https://pendulums.io>

Manage and track your time more efficiently using a modern time tracking tool.

Pendulums is a free time tracking tool which helps you to manage your time in a better manner with an easy to use interface and useful statistics.

Clients: Android, Windows, MacOS, Linux and Web

Server: unclear

The web client (server?) seems to be written partially in Java and not sufficient for most shared hosting. Same time Pedulums seems to be a great software an recommended to put here by Angus McAllister of the FederatedBookkeeping team on Gitter.

## jtx Board journals|notes|tasks

Latest version on F-Droid: 2.00.03-rc01.ose (200030001) submittedt at 2022-10-18

<https://f-droid.org/de/packages/at.techbee.jtx>

GNU General Public License v3.0 only

<https://jtx.techbee.at>

Keep track of journals, notes & tasks - iCalendar compliant and syncable (DAVx5)

- Elevate the power of the iCalendar standard to the next level, use the potential of the combination of journals (VJournal), notes (VJournal) and tasks (VToDo) out of one app and use DAVx5 to synchronize your entries with the CalDAV-server of your choice (coming soon)!
- iCal standard compliant  
Using the iCal standard ensures compatibility and interoperability with other apps and services independent of a dedicated provider or infrastructure. Journals and Notes are compliant to the definition of the VJOURNAL component, Tasks are compliant to the VTODO component. Future features will also include import and export functionalities to and from .ics files 😊
- Combine journals, notes & tasks  
Instead of using separate apps for journals, notes & tasks you can use them out of one hand, combine and link them to each other, e.g. create meeting minutes and link your tasks to them.
- Sync with DAVx5 (coming soon)  
Synchronize your entries with any compatible CalDAV server by using DAVx5. By using DAVx5 you are free to choose your preferred provider for CalDAV, you can even use your local server to store and synchronize your data.  
Note: DAVx5 is an independent app and must be acquired separately.

## Super Productivity

<https://f-droid.org/de/packages/com.superproductivity.superproductivity>

Latest version on F-Droid: 17.0 (17) submitted at 2022-05-03

MIT License

<https://super-productivity.com>

Free to do list & time tracker for programmers & designers

- Organize your daily tasks at one place while making time tracking a lot less annoying. Super Productivity is a ToDo List / Time Tracker / Personal Jira Task Manager for multiple platforms.
- Connect Jira, GitLab and GitHub  
Never miss the important updates on any of your Jira tickets. With our Jira and Github integration you will be notified about ticket changes without having to check Jira or your emails every 10 minutes. Just auto import all tasks assigned to you, plan the details locally and submit worklogs with just a single click.

## Mirakel

**seems to be discontinued after 2015**

<https://f-droid.org/de/packages/de.azapps.mirakelandroid>

GNU General Public License v3.0 or later

<https://mirakel.azapps.de>

Mirakel is a simple but powerful tool for managing your TODO-lists. You can sync your lists with your own server!

### Features:

- |                                     |                                       |
|-------------------------------------|---------------------------------------|
| * Manage your tasks in lists        | * Simple, but powerful User Interface |
| * Tablet-UI                         | * Nice little widget                  |
| * Sort your tasks in different ways | * Fully configurable                  |
| * Notifications & Reminders         | * Easy Backup and Import              |
| * Import your tasks from Astrid     |                                       |
| * Sync with Taskwarrior             | * Sync with CalDAV                    |

## OpenTask

**Latest version on F-Droid: 1.4.2 (82200) submitted at 2021-03-24**

<https://f-droid.org/de/packages/org.dmfs.tasks>

Apache Licence 2.0

<https://opentasks.app>

A task manager app, allowing you to categorise your todo list by urgency, state, timeframe etc.

Tasks can be synchronised with a CalDAV server

using, e.g., `at.bitfire.davdroid` and `com.etesync.syncadapter`

## Invoice Ninja

**Latest version on F-Droid: 5.0.93 (93) submitted at 2022-09-21**

<https://f-droid.org/de/packages/com.invoiceninja.app>

Attribution Assurance License

<https://www.invoiceninja.org>

Create invoices, accept payments, track expenses & time-tasks

Invoice Ninja is an invoicing application that makes sending invoices and receiving payments simple and easy. Our latest version is a clean slate rewrite of our popular invoicing application which builds on the existing feature set and adds a wide range of features and enhancements that you have asked for.

We're similar to WordPress, you can either run the app on our server (hosted) or on your own server (selfhosted) - the server-part is built with PHP/Laravel; many people use Docker or Cloudron to install a packaged version, but you also can install it manually.

## Related

- <https://www.openhub.net/tags?names=timesheet>
- [In / out board](#)
- [Stopwatch for the Timesheet](#)
- [Accounting](#)
- [https://www.drupal.org/project/time\\_tracker](https://www.drupal.org/project/time_tracker)
- <http://directory.fsf.org/wiki/Category/Business/timetracker>
- <http://directory.fsf.org/wiki/Category/Business/timekeeping>
- <https://github.com/jonsmithers/timetracker>
- <https://github.com/digitalkaoz/time-calc>
- <https://github.com/whiteswift/pomodoro-pwa>
- <https://www.dolibarr.org/57-documentation/features/214-timesheets> (good idea to test the app, even if not documented yet)
- [Horde Hermes](#)
  - <https://www.horde.org/apps/hermes>
  - <https://www.horde.org/apps/hermes/screenshots>
  - <https://github.com/horde/horde/commits/master/hermes>
- <https://github.com/wakatime> Open Source plugins in the IDEs to connect to proprietary SaaS
- <https://github.com/overshard/timestrap> in Python
- <https://activitywatch.net/> **Very interesting!**

## alias

---

- [Time tracking](#)
- [Time tracking and invoicing](#)
- [Time Sheet](#)
- [Timesheet](#)