



Scheduled timetable mail

Wise Timetable — one letter per lecturer, their own week, and the school writes the letter



Wise Technologies d.o.o. · Wise Timetable · September 2026

Applies to the 2026 release and later.

Contents

Scheduled timetable mail	4
What a lecturer actually receives	4
The two jobs	4
The screen, field by field	5
The letter, per language	5
Placeholders	6
Trying it before a faculty sees it.....	7
The run summary	7
When the job actually runs	7
What is never on this screen	8
If something is wrong	8
Related manuals.....	9

Scheduled Timetable Mail

One letter per lecturer, their own week, in their own language — and the school writes the letter

Scheduled timetable mail

Every week, before the week starts, each lecturer receives one e-mail: the subject line the school chose, a covering letter the school wrote, a button that opens their own timetable, and their own week attached as a PDF.

Nobody at the school does anything to make that happen. A background job on the timetable server wakes up once an hour, notices that this is the day and the hour that were configured, reloads the timetable from the database and sends the letters.

What changed in the 2026 release is who owns the words. The subject lines, the letters, the send day and the send hour used to live in a configuration file on our machine, and changing a sentence meant writing to us. They now live in the school's own database and are edited on the school's own website, in *Settings* ▶ *Notifications*, by anybody with configuration-administrator rights.

What a lecturer actually receives

Part of the letter	Where it comes from
The subject	The Subject box for that lecturer's language, or the general one
The text	The Letter box — HTML, written by the school
The button or link	The Timetable link template, with this lecturer's own values filled in
The attachment	Their own week, rendered as a PDF at the moment of sending
The subscription address	Optional, written into the letter with <code>{WEBCAL}</code> — see <i>Calendar Subscriptions</i>

The letter is sent as HTML, so it looks like a letter rather than like a text file. Everything a lecturer sees is the school's own wording; nothing is composed by the software except the timetable itself.

One letter per person. Not one letter to a mailing list, and not one timetable for everybody: the run walks the lecturer list, builds each lecturer's own week, expands each placeholder for that person, and sends. A lecturer with nothing in the covered week is skipped rather than sent an empty page.

The two jobs

A school has up to two of these, and they are separate:

Job	Sends	Typically
Next week's timetable	The week that is about to start	Thursday or Friday afternoon
This week's timetable	The week that is running now	Monday morning

Each has its own day, its own hour, its own subject lines and its own letters. Most institutions run the first one only; the second is useful where timetables still move during the week.

The jobs are not created on the website. The configuration file on the server is the only thing that knows a job exists — it names the job and holds the database connection — so the first run of the upgraded server writes the row, and the Notifications screen only ever edits what is already there. If the screen says nothing is configured for your school, that is what it means: ask us, and it fills in by itself.

The screen, field by field

Settings ▶ *Notifications*, one block per job.

Field	What it does
Sending	The job runs, or it does not. Everything else stays as it is
Day	Monday to Sunday, or Every day
Time	The whole hour the job is allowed to send in
Which week	<i>Next week</i> or <i>This week</i>
Send the run summary to	One address that receives a short report after every run
Send one test letter to	See <i>Trying it before a faculty sees it</i> , below

“— unchanged —” is not a blank

Every picker offers **— unchanged —**, and that is the starting value for a school that has just been switched on. It does not mean *Monday* and it does not mean *midnight*. It means **this setting is still answered by the configuration on our server**, which is how every school begins and what makes the change safe: until somebody deliberately sets a day and an hour here, nothing about the school's mailing has changed at all.

Choose a value and the school owns that setting from then on.

The letter, per language

Under each job there is a block per language, and they are read in this order:

1. The lecturer's own language, if a letter has been written for it.
2. **General text**, which every other language falls back to.
3. **Lecturers with no language set** — its own block, because those people are usually the ones an institution most wants to address in English.

Write one for: adds a block for a language that has none yet. Adding a block does not create anything; it draws one more set of empty boxes. A language exists once there is something saved in it.

Remove this language is a tick, not a button, and it is cleared at the save you were going to make anyway — so it can be un-ticked before it takes effect. The general block has no tick: it is the fallback every other language reads, and removing it would quietly empty the letters of everybody you never listed.

Preview the saved letter shows what is in the database, not what is in the box above it. Type, save, then look. The preview is deliberately sealed off — it can draw and do nothing else — and it is not a perfect rehearsal either, because a web browser and Outlook do not render HTML the same way. That is what the test letter is for.

Placeholders

Anything in this list may appear in the **Letter**, in the **Subject**, or inside the **Timetable link**, and it is replaced for each lecturer as the letter is built.

Placeholder	Becomes
{TUTORNAME}	The lecturer's name
{TUTORCODE}	Their code, where the institution uses codes
{TUTORID}	Their internal identifier — this is what a link needs
{LANG}	Their language code, for a link into the right language of the site
{WEEK}	The number of the week being sent
{DATE}	The Monday of that week, as 2026-09-14
{LINK}	The whole timetable link, already assembled
{WEBCAL}	A subscription address for this lecturer's own calendar

{LINK} is the one to put behind a button. The others are for writing a sentence — “Dear {TUTORNAME}, here is week {WEEK}, starting {DATE}” — or for building an address of your own.

The timetable link

The **Timetable link** box holds a template, not a finished address:

```
https://wise-tt.com/web/yourschool/?lang={LANG}&t={TUTORID}&date={DATE}
```

With {DATE} in it, the link opens on the week the letter is about rather than on whatever week the reader happens to open it in. Without it, a letter read on Tuesday shows Tuesday's week. It is worth having.

{WEBCAL} — the one-line upgrade

{WEBCAL} expands to a **subscription** address for that individual lecturer, built from the same timetable link, and offered to Outlook, Google Calendar and Apple Calendar as something to subscribe to rather than to download.

One sentence under the button — “Prefer this in your own calendar? Subscribe once and it keeps itself up to date.” — turns a weekly letter into a calendar that is never stale. The letter still goes out; the calendar stops being a photograph. *Calendar Subscriptions* is the manual that covers what a subscription is and how often it refreshes.

Trying it before a faculty sees it

Send one test letter to is the field to use before every wording change goes out to three hundred people.

Type an address, save, and wait for the next hourly pass. The server then sends **one real letter** — a real lecturer, their real timetable, their real PDF, every placeholder expanded exactly as it will be — to the address you typed, and empties the box again.

So: a filled box means *pending*, an empty box means *nothing waiting*, and clearing the box cancels a test that has not gone yet.

Four things about it are worth knowing, because each of them would otherwise surprise somebody:

- **The website cannot send mail.** It has no mailbox and no timetable data. Typing an address writes a request; the server does the sending.
- **A test never interrupts a real run.** Ask for one during the job's own sending hour and it waits for the next pass. A test that narrowed the real run to one letter would be the worst thing this feature could do.
- **No run summary is sent for a test.** "We checked 700 lecturers" after one letter is a sentence a machine would file and a person would believe.
- **It takes up to an hour.** The job is an hourly pass, not a button.

The run summary

Whoever is named in **Send the run summary to** receives a short report after every real run: how many lecturers were checked, how many letters went out, and — this is the part worth reading — **whether the mail server refused any of them**.

The summary says

It means

Every letter was accepted by the mail server

The letters left the building. Whether a mailbox then filed one as junk is outside our sight

n of those letters were REFUSED

The mail server rejected them. The event log on the server names every address

A refused letter is almost always one bad address in the lecturer list — a typing mistake, or a mailbox that was closed when somebody left.

When the job actually runs

The background job wakes up **once an hour**. On each wake it re-reads its configuration — so a change saved on the website is picked up on the next pass, with nothing to restart — and then asks one question: is this the configured day and the configured hour? If it is, it reloads the timetable from the database and sends. If it is not, it goes back to sleep.

Two consequences follow, and both have caused a puzzled mail at some point:

- **The pass is not on the hour.** It happens once an hour counted from whenever the server was last started, so a job configured for 14:00 may go out at 14:20. It goes out within its hour, not at the top of it.
- **A change saved five minutes before the sending hour is in that run.** There is no separate reload and nothing to restart. There is also no way to recall a letter that has already gone.

What is never on this screen

The mailbox the letters are sent from, its password, the mail server, and the database connection are **not** on the Notifications screen and never will be. They stay in the configuration on our server, where a school's configuration administrator neither needs them nor can reach them.

That is also the rollback: because the file still holds everything the screen does not, a school can be put back exactly as it was by restoring its block there, and the website stops mattering for it.

If something is wrong

What you see	What it means
<i>"This screen is not switched on for your school yet"</i>	The school's mailing tables do not exist yet. Ask us; it is one upgrade of the server, not a project
<i>"No mailing job is configured for this school"</i>	The tables are there but no job has run yet to write its row
Some of what you typed was not saved	The server could not use those rows and names them. Almost always a language block with nothing valid in it
A saved change made no difference to the letters	Check Sending , then check whether the pickers still say — <i>unchanged</i> — the file is still answering
The letters go out an hour later than configured	That is the hourly pass, counted from when the server started. Within the hour is correct
The test letter never arrived	It goes on the next hourly pass. If the box has emptied itself, it was sent — look in junk, and look at the run's log
A lecturer got no letter	They had nothing in the covered week, or their address was refused. The summary and the log distinguish the two
The preview looks right, Outlook looks wrong	Outlook renders HTML with Word's engine. The preview is a browser. Use the test letter as the judge, never the preview
The link opens the wrong week	The link template has no {DATE} in it

Related manuals

- **Calendar Subscriptions** — what {WEBCAL} gives a lecturer, and why a subscription beats a downloaded file.
 - **Online Meeting Links** — how the join address gets into the timetable, and therefore into the letter and the calendar entry.
-

Wise Technologies d.o.o. · Ljubljana, Slovenia · wise-tt.com