

You as a UX and GUI expert pls help me design the frontend pages for our AI driven Event Request section where I connect our AI agent mail workflow backend to. We are working on the current branch and should use existing pages and first analyse and then review if the UX is convenient for an event manager. To answer that we have to know better and research where they come from, what their habits are and what the SOTA standard solution design if existing for our case would be (do a web search!). Then I also need the corresponding setup page for that with all settings clearly understandable for event managers what each option does (small description such as here in the already implemented calendar settings on general settings s.t. it's clear for existing and also new event managers what the setting does exactly, keep the language simple and with as few words as possible: quality over quantity and use words which are daily routine of event managers and are known to them

Example for description style (currently the rest of the setup page is designed like this :

Calendar Settings

Default View

Choose your default calendar view mode

Calendar View

Default Calendar

Choose how your calendar displays by default

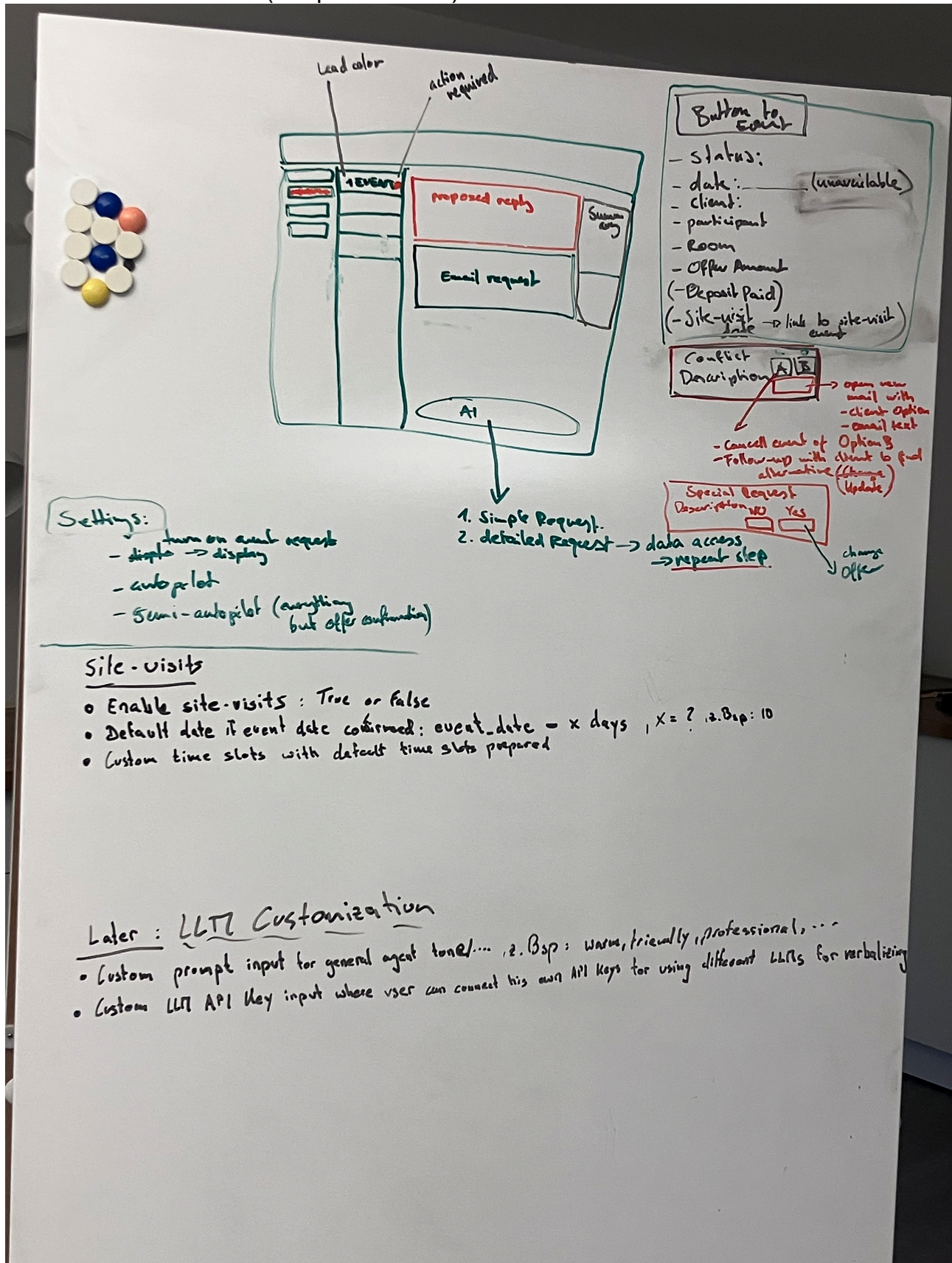
Month

Week Starts On

Choose the first day of your week

Sunday

First take, it shows also what needs to be in settings pages, ignore the LLM customisation for now (not part of MVP)



New Design (only idea exists yet): S.t. event managers / users can understand what the AI did we need some kind of "action list" that shows the most

important AI actions (executed on the backend) which affects the event managers GUI in any way : check out the frontend and understand it, see also **FRONTEND REFERENCE**, eg updated calendar: changed time for site visits from ... to ... , special request: Projector added , checked date availability: show that or not ? Maybe thats not very useful: rather show either something like : Failed to book room on date ... (already booked) ,

Created offer for client (name),

Updated offer for client (name),

Updated event on calendar: Changed date of ... from To (Detours)

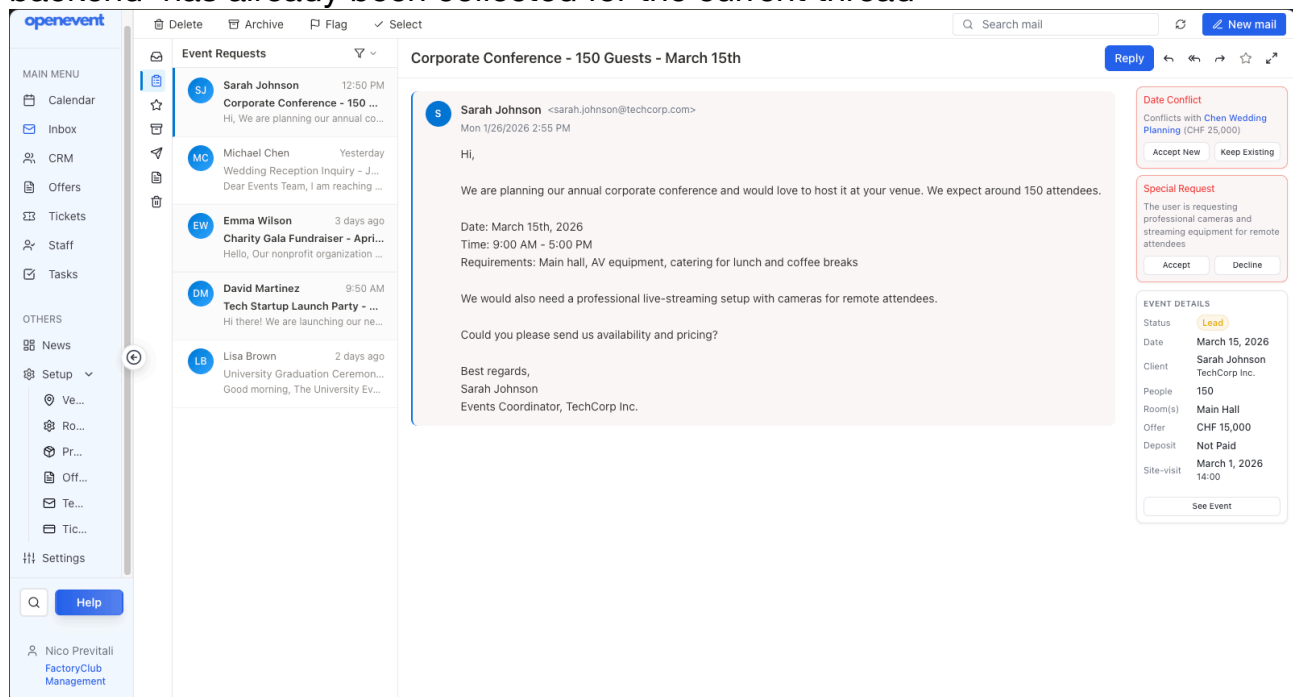
Added CRM entry: New client (name)

Updated CRM entry for client (name) : changed billing address (or also added billing address if it happens for the first time)

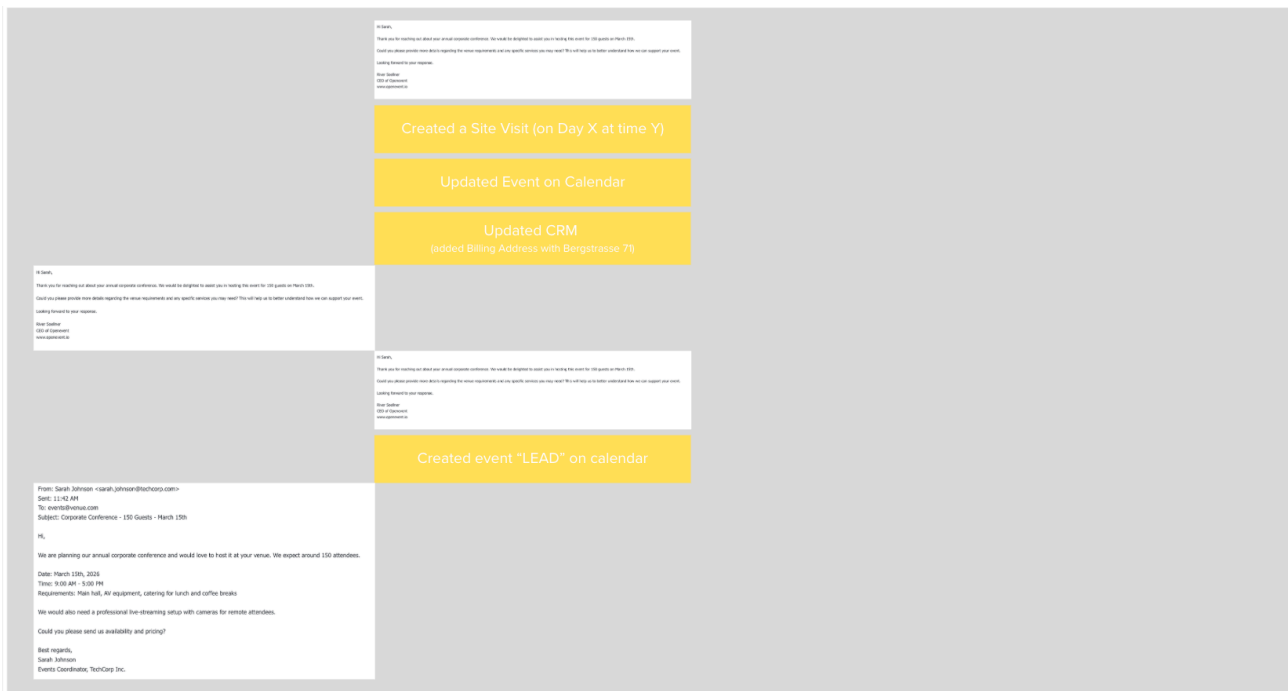
Client Update: Changed

1.) Design the Event Request Inbox

Current GUI for the event workflow: first go to tab below the mail inbox named “event request” , then you see all your client messages in this box , on the left there’s 2 boxes already designed which pop up if one of these actions occurs: either date conflict or special requests and there’s also a newly designed box as some kind of progress bar to keep track what most relevant client information/ backend has already been collected for the current thread

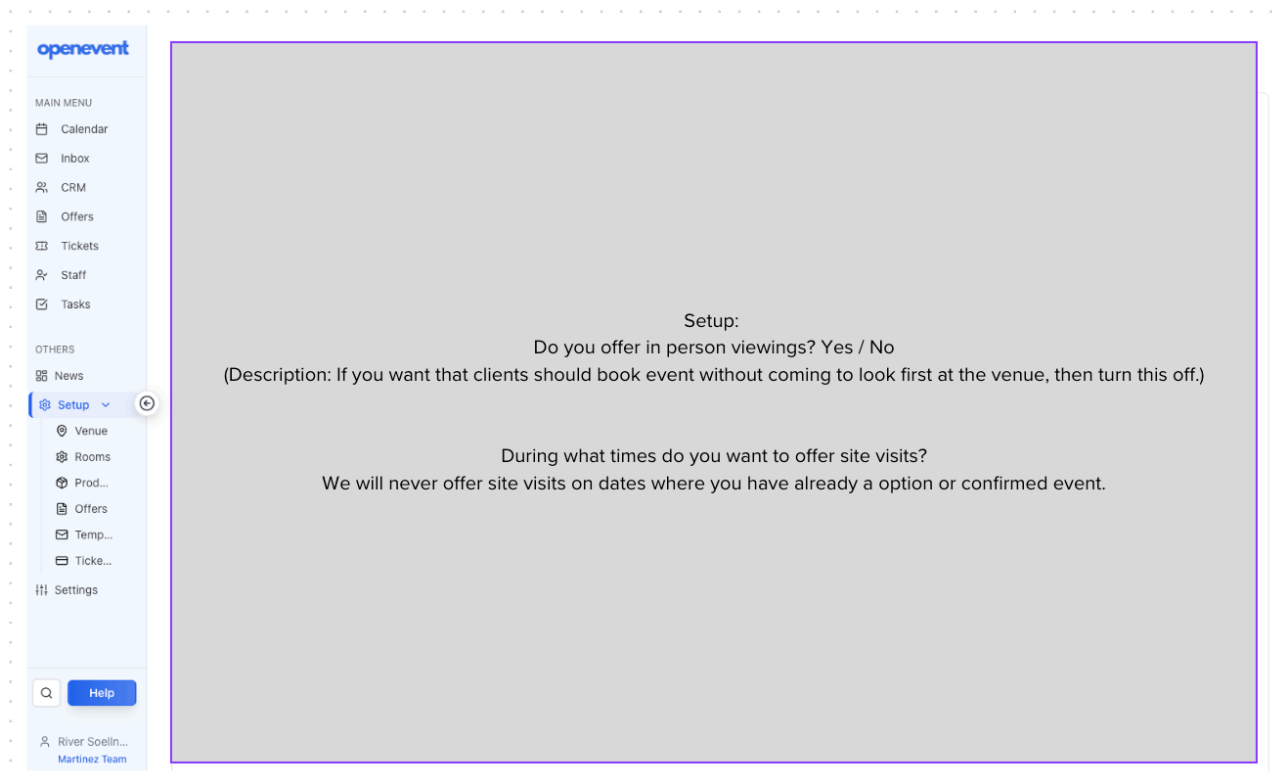


Action list idea (yellow box) : this is only for brainstorming - the GUI isn't decided yet



So how to design this optimally using sota event managers etc. , also vocabulary must be clear and understandable for the target audience , also there must be no redundancy with the rest our websites: we dont want the same thing twice ,

2.) Create and Design the Setup Page (Settings) for the Event Request feature



Explanations must be clear/complete and everything we need for implementation clearly mentioned (also menu dives etc : don't shortcut that on our requirement list) ,

Based on web search , the analysis of the frontend (everything) , the images wrt to the new frontend features and my instructions review our current state and ideas and create a design plan for these 2 tasks (no implementation details yet , just all the relevant UX questions and answers we need for this , describing the whole outlook etc for each of the 2 pages.

Related documents:

- in docs/integration/frontend_and_database : from manager/frontend view it's docs/integration/frontend_and_database/ mainly:
MANAGER_INTEGRATION_GUIDE.md , docs/integration/frontend_and_database/specs , docs/integration/frontend_and_database/specs/EMAIL_WORKFLOW_INTEGRATION_REQUIREMENTS.md
- docs/integration/frontend_and_database/specs/Frontend_REFERENCE.md