

|  |
| --- |
| Software Requirement Specification |
|  |
| November 16  WAVALE  Authored by: Yannick K. |

# Introduction

|  |
| --- |
| Here is a trimmed down document for the Subscription Webapp called Invoiceally: It is intended for developers, designers, and testers working on “Invoiceally” as well as project investors. This plan will include a summary of:   * how the system will function * the scope of the project from the development viewpoint * the technology used to develop the project, and * the metrics used to determine the project’s progress * Overall Description   Companies/Freelancers need remote [financials tools](https://relevant.software/blog/10-best-tools-for-your-remote-product-development-teams-sync/), especially now that more people are working from home. The problem is that most companies or freelancers end up using multiple applications to accomplish this: one for invoices, one to track Project/Tasks, and one for time tracking. “Invoiceally” will solve this problem by combining these features in one application. *Customers* The customers will be freelancers & enterprise companies. It’s will target the general public. *Functionality*  * Users should be able to sign up with enterprise LDAP accounts. * Users should be able to sign up with email, and socials accounts (Google & Facebook) * Users should be able to create Company organization (Business profile) comprising admin & sets of users with predefined roles. * Users should be able to track Time for each project /activities/Tasks. * Users should be able set a billing schedule * The application should be able to handle unlimited users at a time. |
| *“Time Tracking & Organizational features references are and.co & Elorus.com”* |

### *Platform*

The application is based on PHP, developed on CodeIgniter to enable the creation of a web-based application, and later 🡺 IOS mobile app, and an ANDROID mobile app (future Roadmap)

Current Webapp invoiceally.com is based on a script bought from Code canyon : <https://codecanyon.net/item/accufy-saas-financial-software/25039709>.

Installation was done on hosting Bluehost.com. No modification has been done to the current script.

Hosting is a shared hosting.

### *Development Responsibilities*

The developer assigned to this project will be responsible for:

* Updating the current code for the application with new features identified, update the database, and managing release after the new configs are implemented.
* Fix current identified bugs or configs
* Fix any bugs that may arise following upgrade.

### *User Class and Characteristics*

There will be a class of users called:

1. Main admin that will have permissions to access all functionality of the app, including:

* Creating /deleting users/company
* Giving access to specifics features to the entire company or individuals depending on the chosen subscription.
* Creating New categories
* Add New invoices template
* Backing up entire application.

1. Standard users will have access to all functionality of the app except admin identified one.

2.1 Users that create individual account

They will have access to Webapp functionalities under individual account, meaning

* Creating/sending/checking/deleting/modifying invoices/estimates/expenses/contracts
* Track time
* Create project, activities, Tasks
* Add new user to participate.

2.2 User that register as company/organization will have different sets of roles:

* + 1. Account Manager Role: will have access to all functionalities of the app under their account except Main admin identified one. They will perform account management (manage users, organization settings and the Invoiceally subscription) & financials (expenses, invoices, estimations)
    2. Roles with limited or specific access to functions:
* **Leaders:** Leaders can perform any action, **except from** account management (cannot manage users, organization settings or the Invoiceally subscription). They are allowed to send invoices/Estimates, Contract & check the expenses.
* **Project Managers**: Project managers can track time for projects they're a member of. They create task, activities and can assign it to the Buddies. They can check the expenses for the project. They **may be allowed** to manage projects & other people's time on a per-project basis. They have access **only** to the financial data of projects they manage but they **cannot** see project costs. They are allowed to send invoices/Estimates
* **Buddy**: They are employees will be able to **only track time** for projects they're a member of. They can create activities on which they are working on. They are not allowed to send any invoices/Estimates.

### *System Features*

#### Functional Requirements

* Users should be able to create ad hoc chat groups comprising sets of users and send private messages to other users.
* Users should be able to have text chats that they can break into threads.
* Chats should be able to be archived indefinitely so users can reference past chats.
* Users should be able to upload files to chats for reference.
* External Interface Requirements

#### User Interfaces

* Front-end software: Java, HTML, CSS, JSON
* Back-end software: PHP
* Database software: MySQL
* Software Framework: CodeIgniter

### Performance Requirements

* The application should load and be usable within 3 seconds
* The application should update the interface on interaction within 2 seconds
* The database should be normalized to prevent redundant data and improve performance
* The database should be distributed to prevent outages

### Safety Requirements

* Databases should use sharing to be redundant to prevent loss of data.
* Backups of the databases should be done hourly and be kept for one week – Check with options are provided by the hosting.

### Security Requirements

* Any keys used for the REST API should be stored securely 🡪API On the future Roadmap, not needed currently
* Only the REST API should be able to connect to the databases🡪API On the future Roadmap, not needed currently
* Databases should be behind a firewall 🡪check if this is handled by the hosting
* Webapp should be safe and protected

### Software Quality Attributes

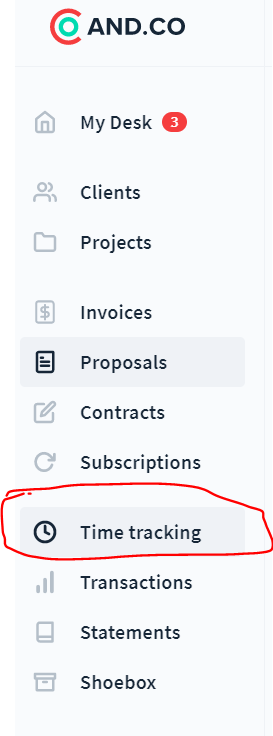
* Availability: application is critical to the business, we will have a goal of four nines(99.99%) availability.
* Correctness: The application should never allow anyone to access account or discussions not intended for that user or another company.
* Maintainability: The application should use continuous integration so that features and bug fixes can be deployed quickly without downtime.
* Usability: The interface should be easy to learn without a tutorial and allow users to accomplish their goals without errors.

### *Application Features*

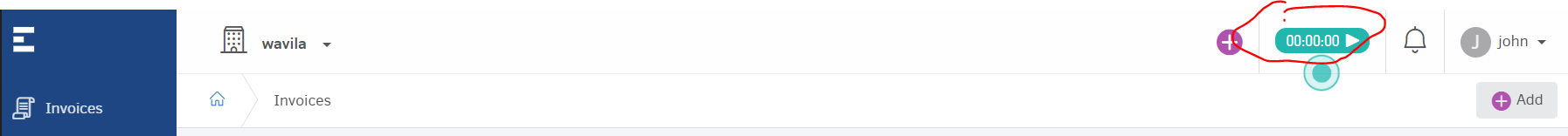
* Time Tracker

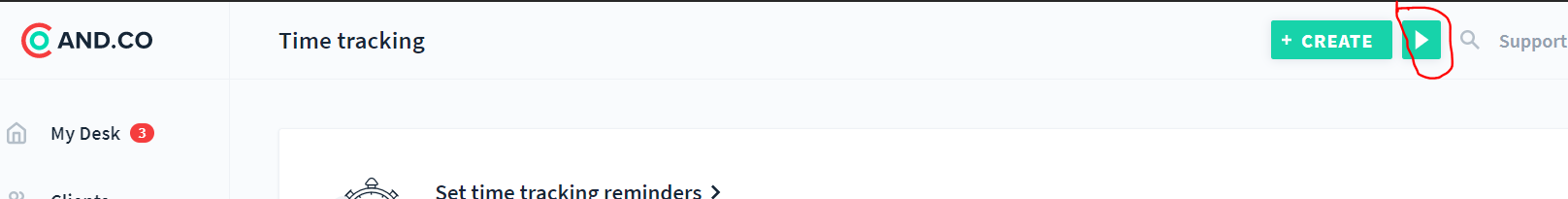
Target🡺Add a time tracker module to track activities, tasks and be able to invoice it or send it through proposals.

* Time Tracker Section should be created on the left (Reference And.co)

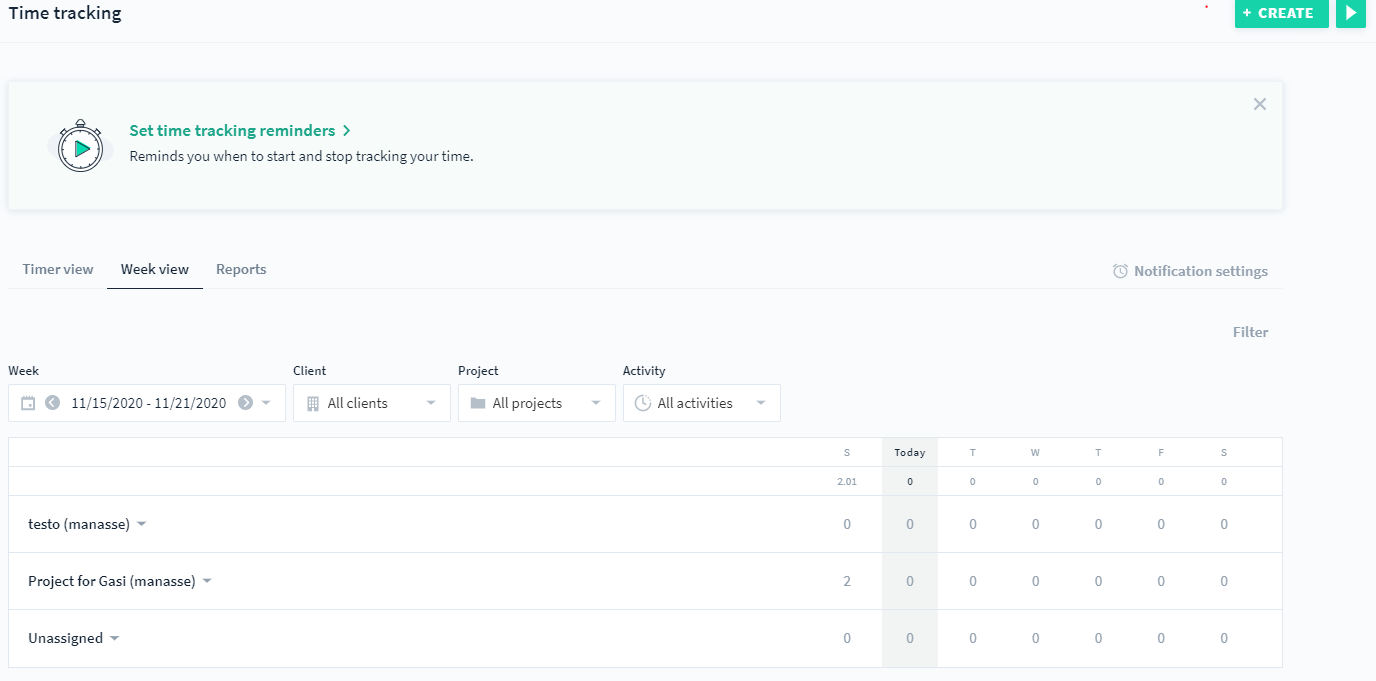


* Time Tracker should be live and available from every screen (Reference and.co & elorus)



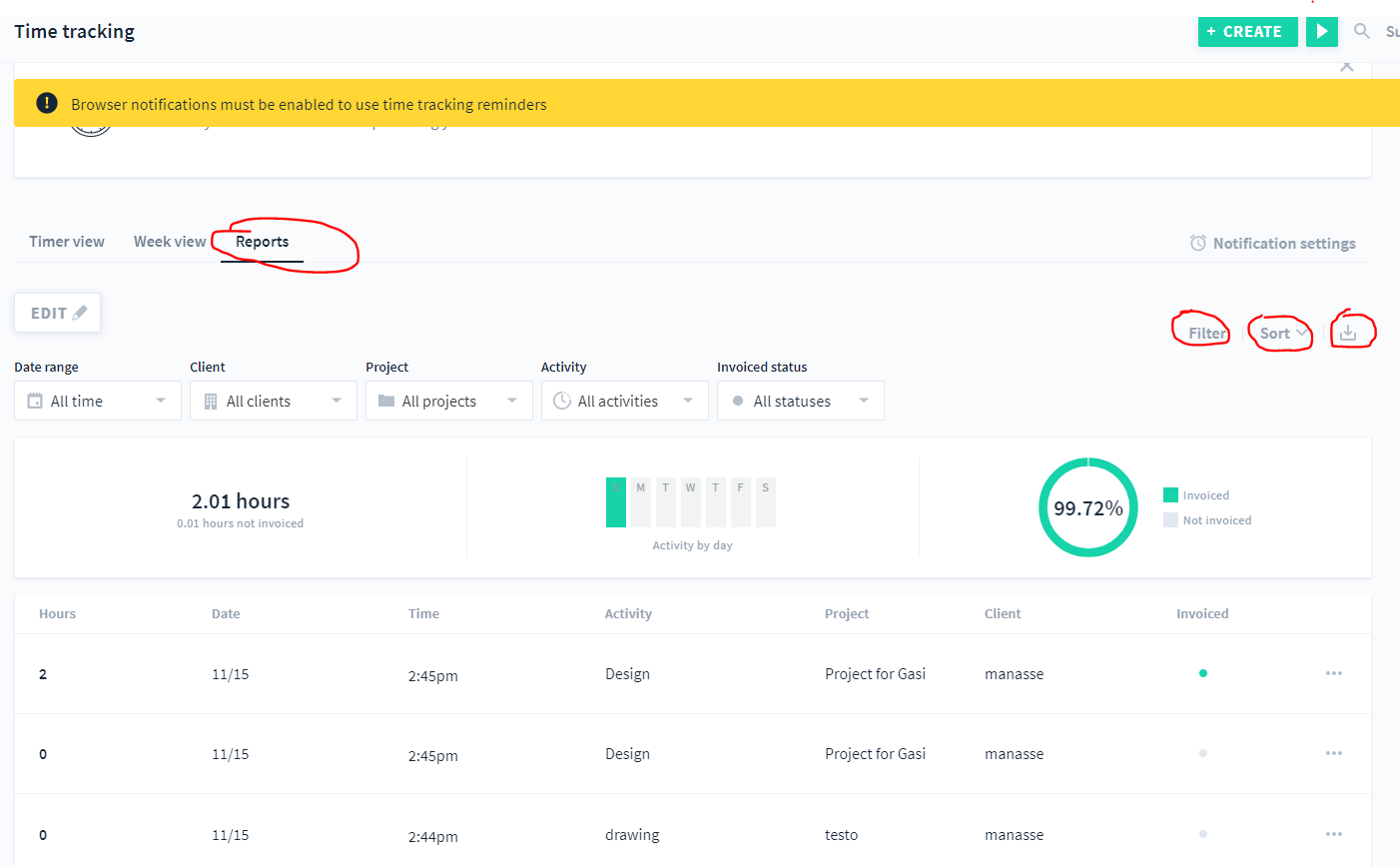


* Time Tracker interface should look like And.co and follow the same logic & workflow process



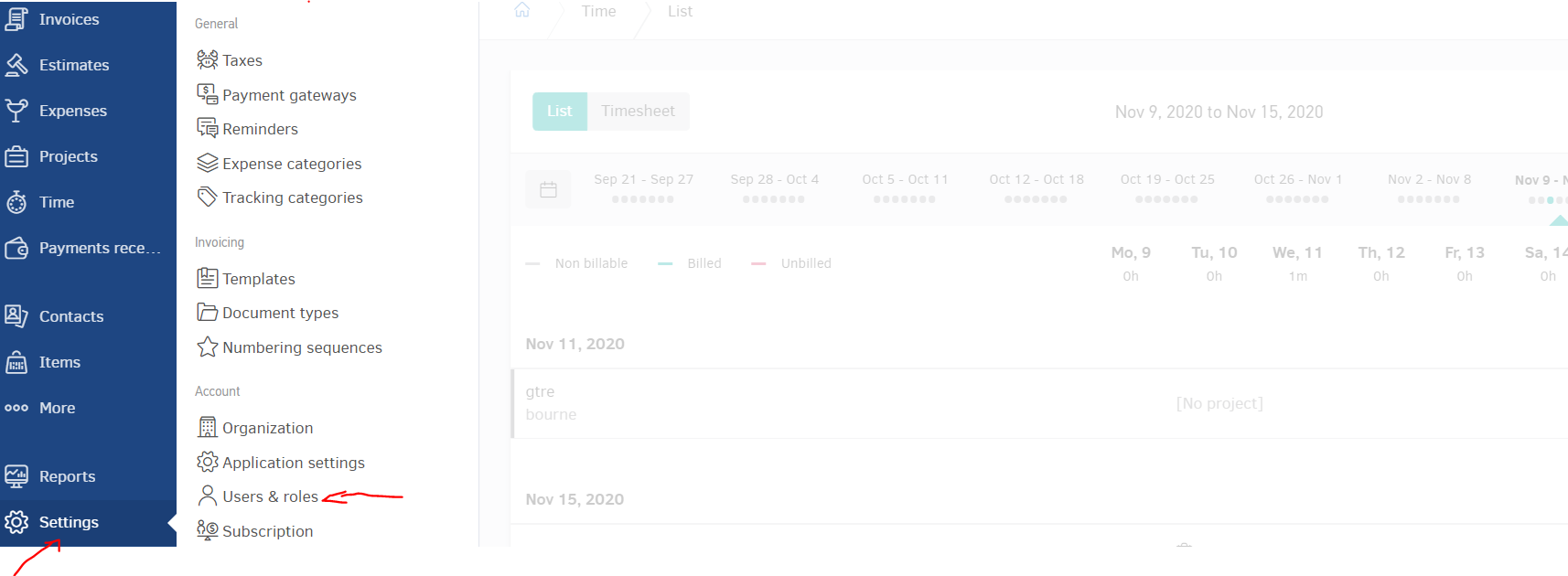
* Time tracker is linked to activities, Tasks and projects hence its important to get useful insights for your projects & team's performance through the time tracker. Report of working time should be accessible (daily, weekly, monthly view) and downloadable (Export)

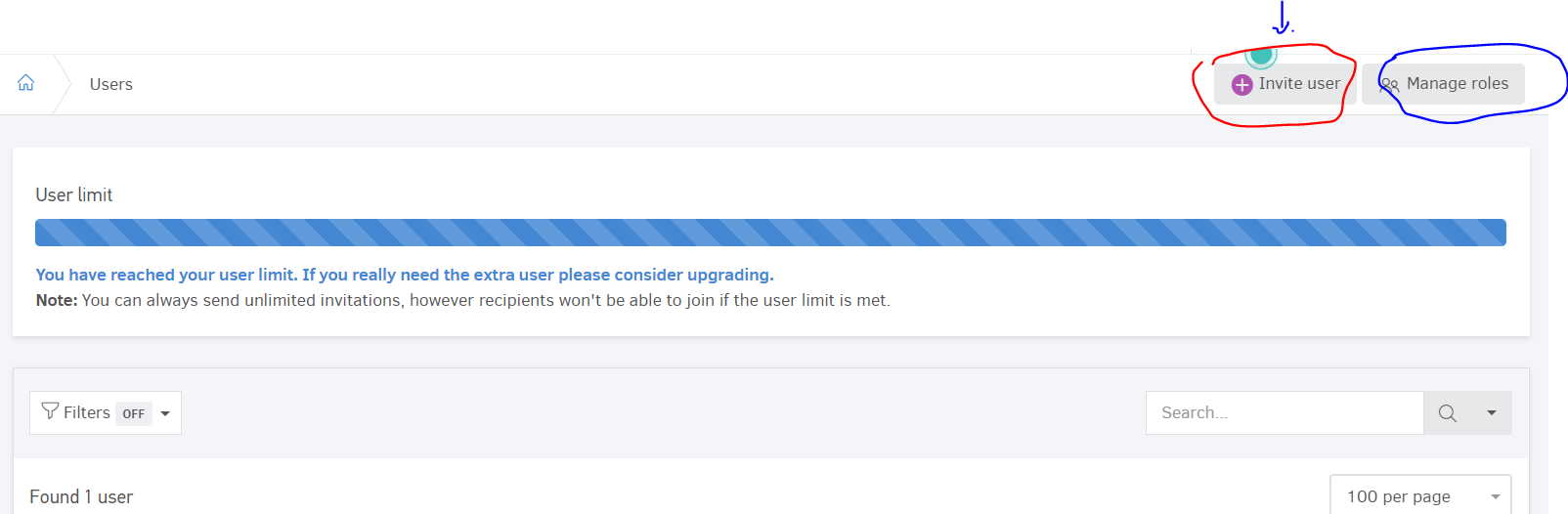
Quick question for thought: What if the user has internet issue, is the time stopped? Save at the moment internet was cut and where is it save? Is the timer resume at the same point when internet is back?

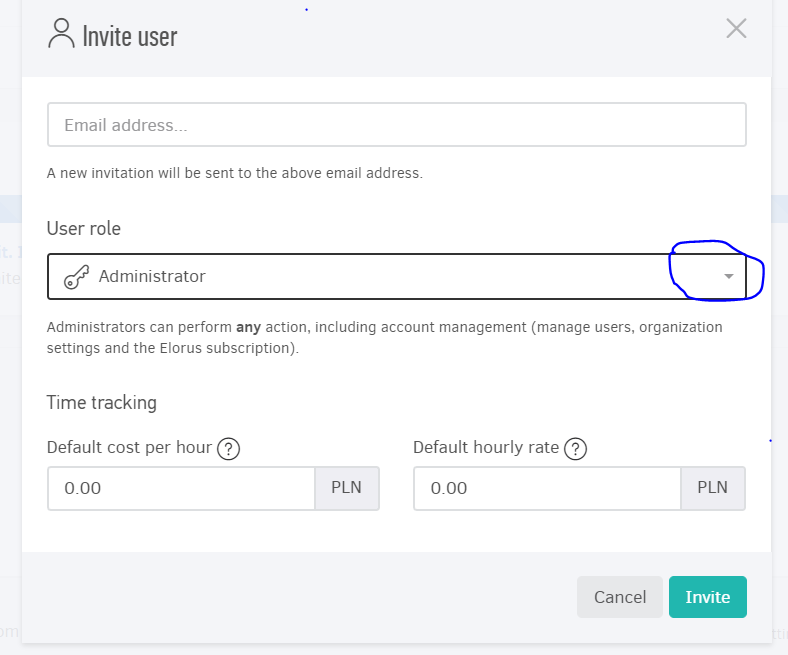
* 
* Organization

Target 🡺 Need to have a company module for user who register as company who need to have several sub-users with different access and profile.

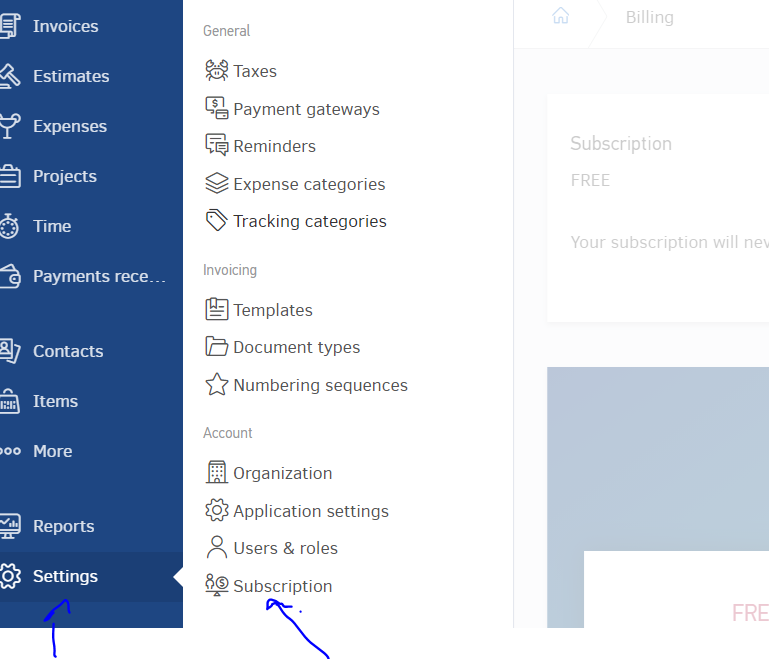
* User should be able to choose during registration option to register as company or Individual
* User should be able to choose how many sub-users (0 to 100) will be accessing post registration
* Post registration user (Account Manager ) should be able to set /manage the role of the subs users and invite them to sign in

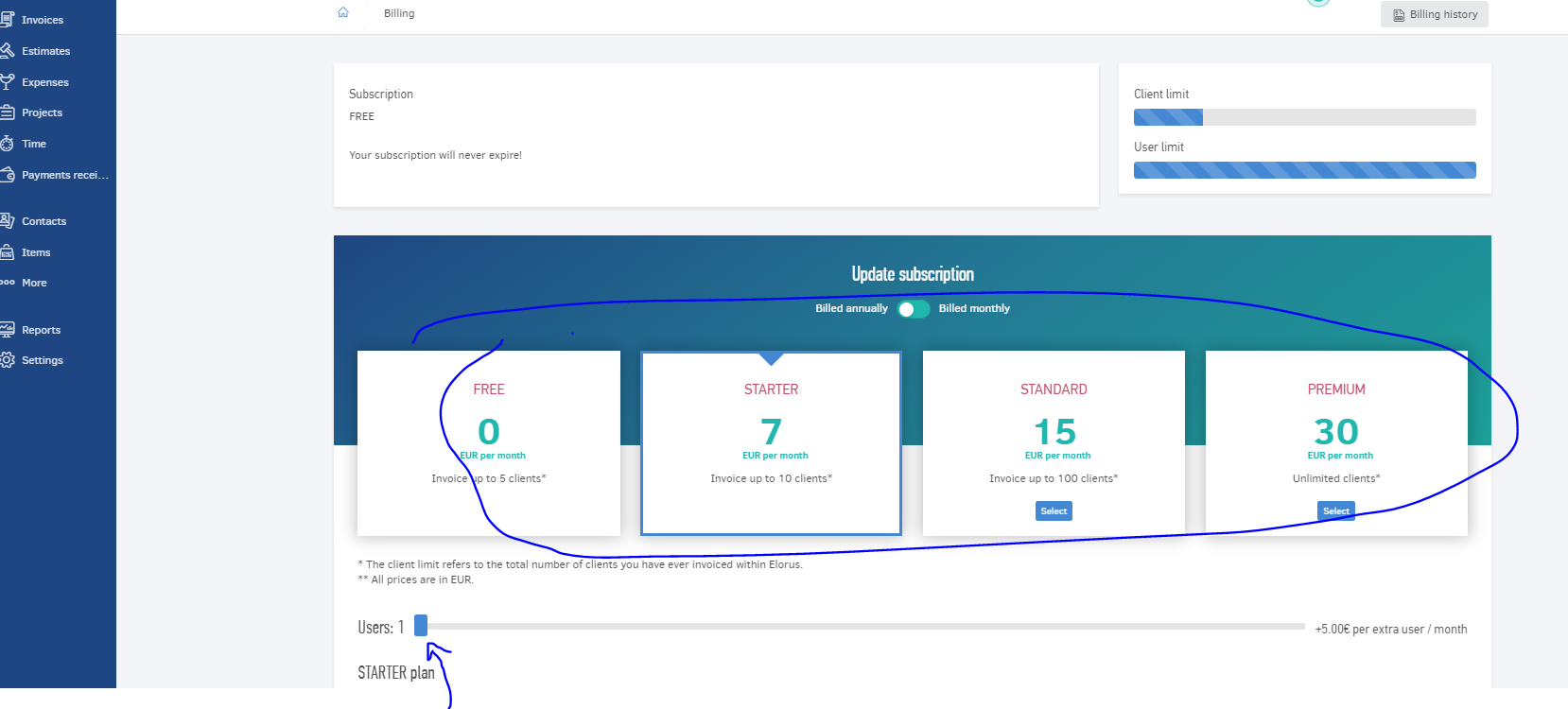






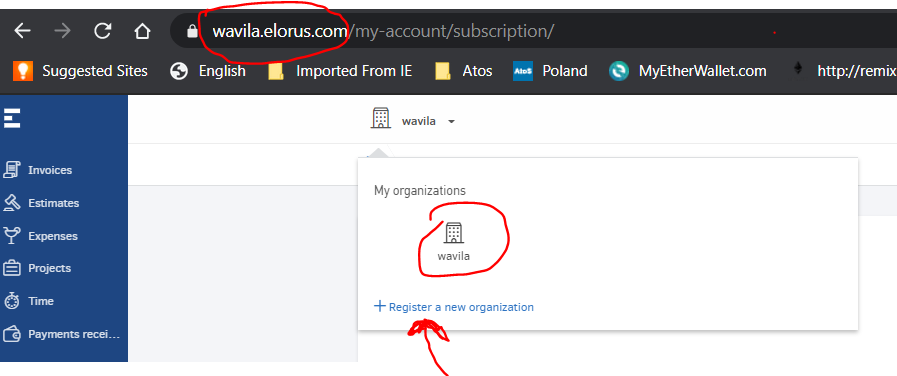
* User (individual/company) should be able to invite sub-users (0 to 100) post registration or upgrade the subscription





* User who open company profile should have the private link where they (include its subs-users) can access. Of course, he should have the opportunity to create new company directly from the same window

Ex: User open nike account on invoiceally.com 🡺 his new link will be Nike.invoiceally.com

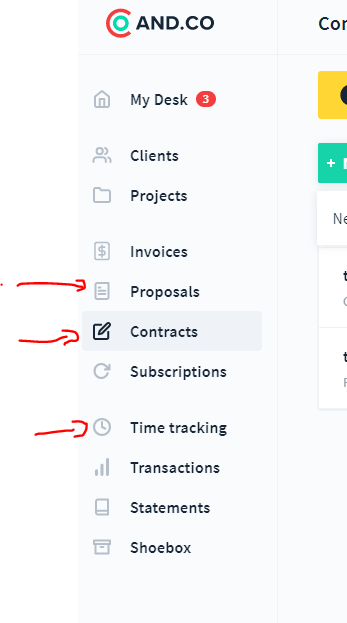


* Minors Update

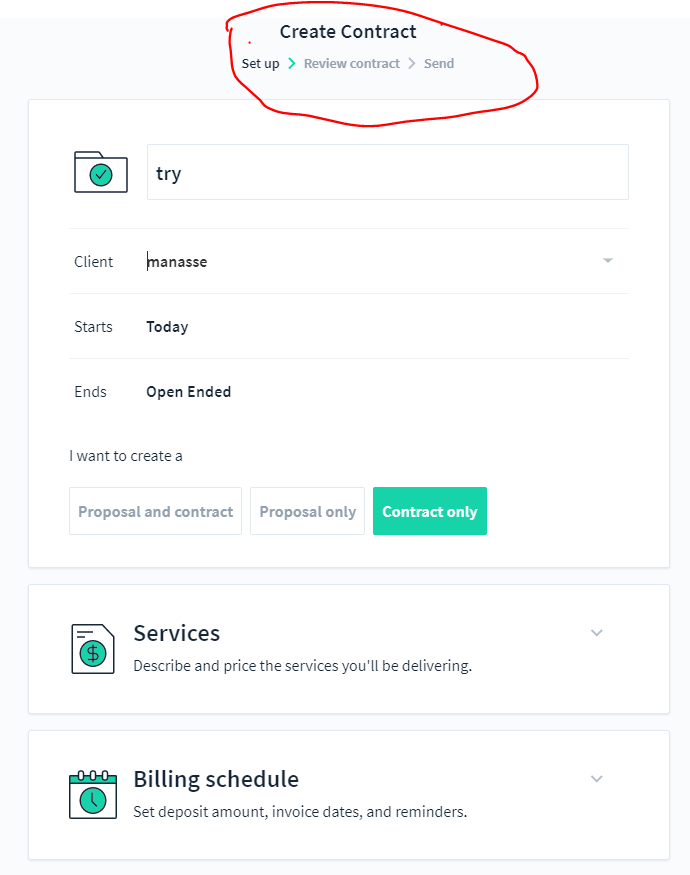
- Welcome Email after Sign up and validation.

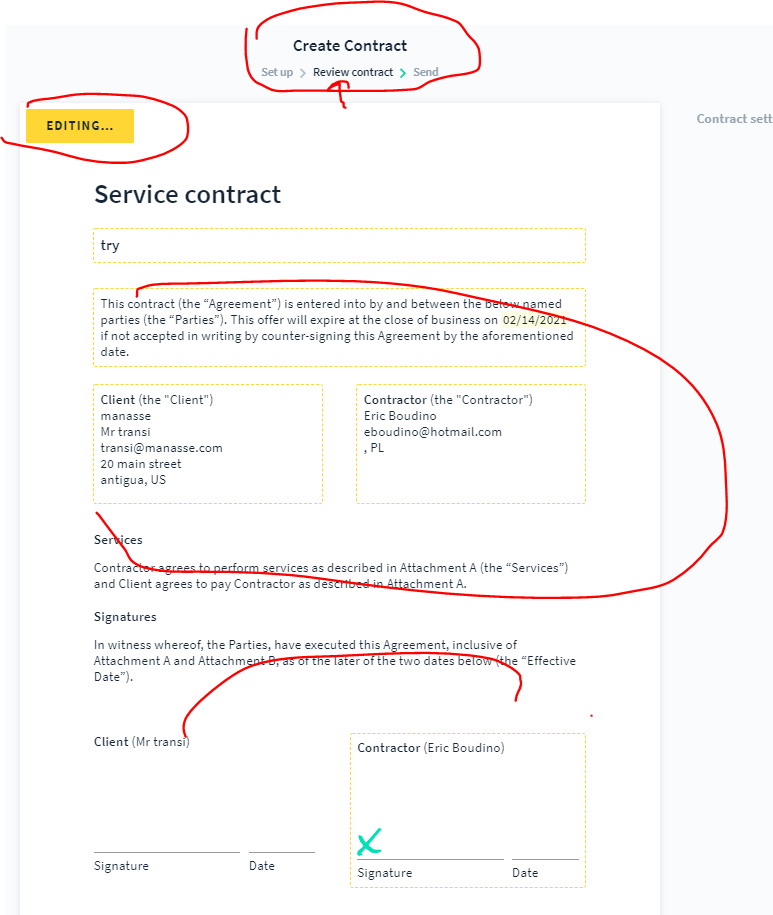
- Sign up by social media facebook and google

- Add Contracts, Proposals and Time Tracker Section in the left panel side (Reference And.co)

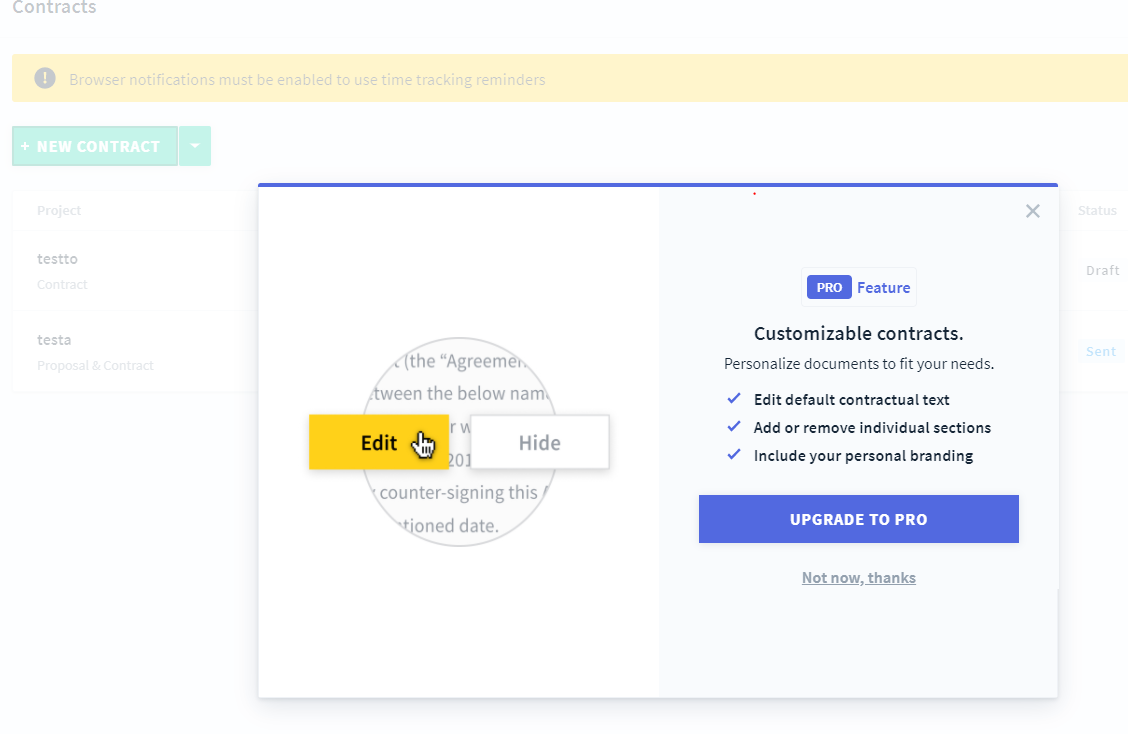


- Add Contracts and Proposals logic & workflow as seen in And.co, Integrate a modifiable contract model that will be updated by users (premium feature). Insert billing schedule in invoicely.com (contracts, Proposals, invoice and estimates) as seen on and.co

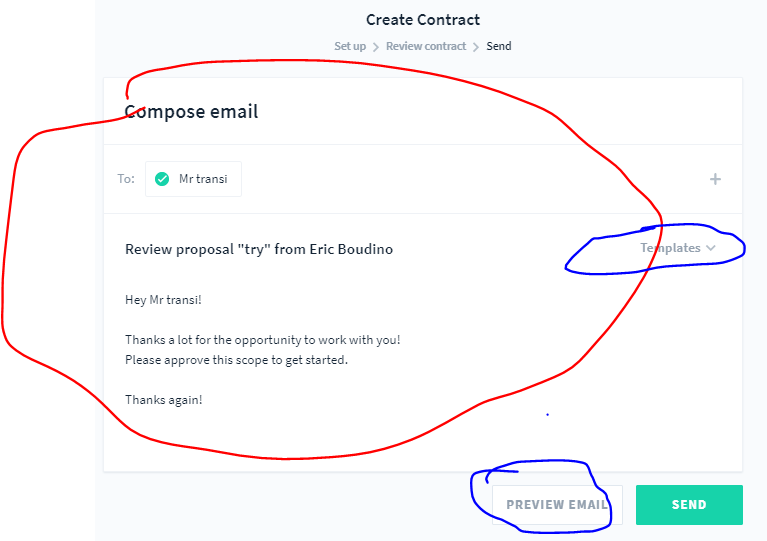




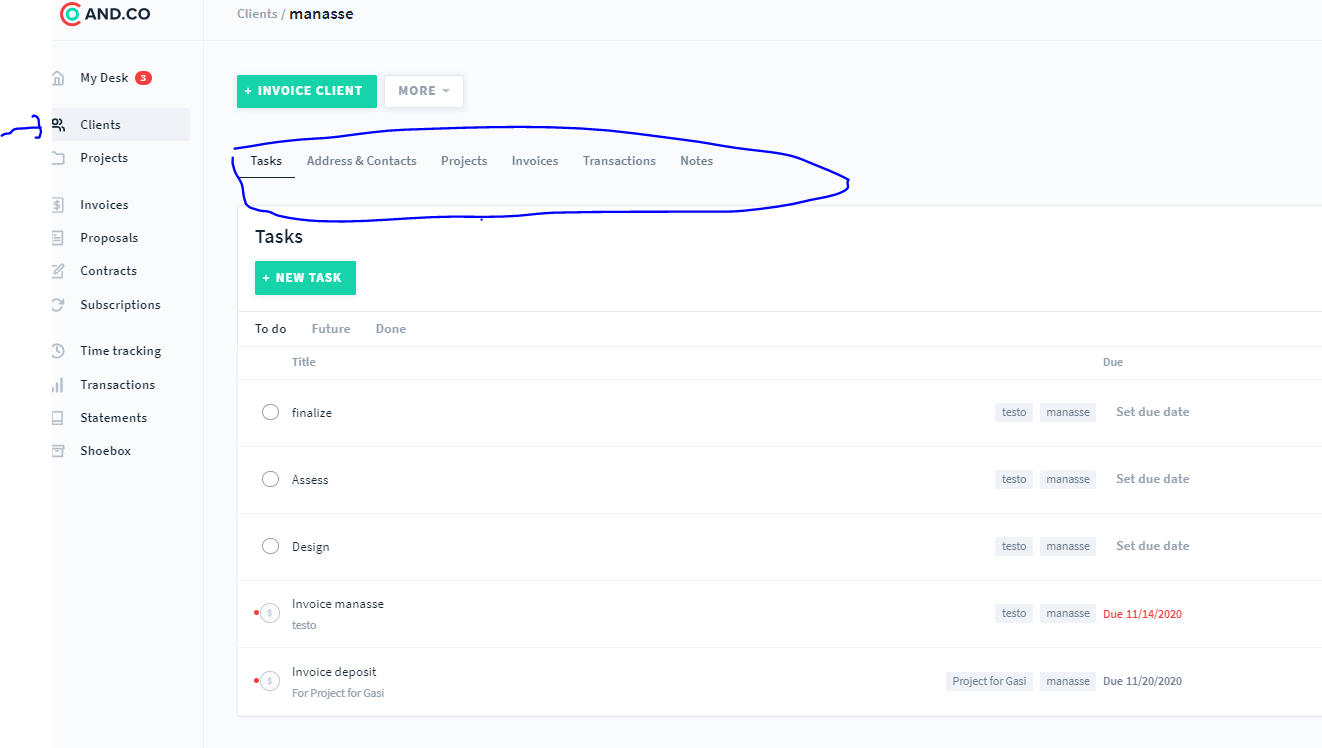
- Add a pop up to upgrade to pro version for those who choose lesser subscription



* Configure email settings to send invoice, proposals, Estimates directly from the app



* Modify and improve the clients view: add a 3rd view as discussed (Reference And.co)



* Modify and improve the project view: add a 3rd view as discussed (Reference And.co)

