Training guide
Odoo 19

Project in Odoo 19

The Project app from zero to one hundred: tasks, stages, milestones, timesheets and profitability, with real cases, strengths and weaknesses, and a 50-task walkthrough of an Odoo deployment.

1–3	Understand	what a project is, the building blocks
4–5	Set up	configuration, views, connections
4–6	Connect	sales, timesheets, accounting, HR
7–11	Deploy	50-task walkthrough, cases, checks

Written for a new team member who has to learn Odoo projects fast: it starts from the idea, then moves to the Odoo screens, with Iranian and international equivalents side by side.

Powered by **Amir Akbari**

October 2026

The short version

The Project app is where work is planned, split into tasks, handed to people, tracked to done, and turned into hours and money. Used well, it is the shared to-do board, the timesheet and the margin report of a team, in one place.

This guide is for a new team member who has to learn the Project module quickly. It starts with what a project is and why teams use one, then explains the building blocks, the set-up, the views and the connections to Sales, Timesheets, Accounting and HR. It gives an honest list of strengths and weaknesses, real-world cases, and a walkthrough of 50 tasks in an Odoo deployment, written so that a reader can follow them one by one. Two tags mark the difference between a company in Iran and one elsewhere: Iran and Outside Iran

#	Step	What you decide	Can it change later?
Set-up			
1	Enable features	Tasks in project, milestones, subtasks, dependencies, recurring tasks, timesheets, profitability	Yes
2	Stages	The columns a task goes through: New, In progress, Review, Done	Yes. Renaming is safe; removing a used stage moves its tasks
3	Projects	One per client, product or internal initiative; privacy and followers	Yes. Archive, do not delete
4	Tags, priorities, templates	Labels, urgency and reusable task lists	Yes
Work			
5	Tasks and milestones	The work broken down, owned, dated and tracked	Yes
6	Timesheets and profitability	Hours recorded, costs and revenue compared	Yes, but past hours carry their cost
7	Connections	Sales orders, invoices, accounting analytics, HR	Yes
Check			
8	Go-live tests	Proof that tasks, hours and money agree end to end	

The one idea that makes the rest predictable

A project is a box, a task is a card in it, and hours are the bridge between a task and money. If a task has an owner, a stage and a deadline, and its hours are recorded, Odoo can tell you where the project stands, what it has cost and what it is worth. If any of those three is missing, the report is only a guess.

What is in this guide

1 What a project is, and why a team needs one

2 The building blocks of the Project app

3 Setting up Projects

4 Working in the views: kanban, list, calendar, Gantt

5 Tasks, milestones and dependencies, in depth

6 Connections: Sales, Timesheets, Accounting, HR

7 Strengths and weaknesses

8 Real-world challenges and how Odoo solves them

9 Iran and the rest of the world on one page

10 A 50-task deployment walkthrough, explained simply

11 Profitability and reporting

12 Go-live checks and common mistakes

13 Cheat sheet, glossary and self-check

Scope: Odoo 19, Community and Enterprise. Features marked Enterprise need Enterprise, notably the Gantt view, Project Updates and Planning. Menu labels can move between 19.x releases; where a path looks different on your screen, search the menu for the name of the screen. This guide follows on from *Inventory in Odoo 19* and uses the two sister companies: Sahand Industrial in Tehran and Sahand Trading in Dubai. The client examples are fictional.

About the company examples

Some examples are named after well-known companies, such as Toyota, Amazon, Airbus and Shopify, because the kind of project problem they are known for is easy to recognise. This guide does not say that those companies use Odoo. It shows how the same problem is solved with standard Odoo features.

1 What a project is, and why a team needs one

A project is a piece of work with a goal, an end, people and money. This section assumes you have never managed one in software.

A kitchen renovation, in Odoo words

Suppose a family wants a new kitchen. Somebody must list the jobs (remove the old cupboards, lay the pipes, fit the counter, paint). Each job has an owner, a date and a cost. Some jobs cannot start until others finish: the counter cannot be fitted before the cupboards are in place. At the end, the family wants to know whether the budget held. A project tool does exactly that, for any kind of work.

Kitchen renovation	Odoo Project
The whole renovation	**Project**: one box with a name, a customer, a stage and a manager
Each job: remove cupboards, fit counter	**Task**: a card with an owner, a deadline and a description
“Kitchen is ready for the family”	**Milestone**: a dated checkpoint that means something to the customer
“Counter after cupboards”	**Dependency**: a task that cannot start until another finishes
Workers and the hours they spend	**Timesheet**: hours recorded against a task
Price the family pays, costs paid out	**Profitability**: revenue and costs of the project compared

What a project tool is for

Planning
Break the goal into jobs, order them, and see the dates before work starts.

Responsibility
Every job has one person who answers for it.

Progress
The board shows what is waiting, in progress, in review and done.

Time
Hours are recorded where the work is done, not reconstructed at month end.

Money
Revenue, costs and margin are visible while the project runs.

Communication
The client sees progress through a portal, without emails.

Six words you will meet

Word	Meaning	Example
Project	A container of tasks for one goal	Alborz Switchgear Installation 1405
Task	One unit of work, with an owner and a stage	Install panel 3 at Alborz site

Word	Meaning	Example
Stage	The column a task sits in on the board	New, In progress, Review, Done
Milestone	A dated checkpoint, usually linked to a deliverable	Site handover, 15 Azar 1405
Subtask	A task inside a task	Wire panel 3, test panel 3
Timesheet	A record of hours spent by a person on a task	6.5 hours by Reza on Tuesday

Who uses the Project app

Role	Does	In Odoo
Project manager	Creates the project, sets stages and milestones, watches progress and margin	Project manager access; the project dashboard
Team member	Works the tasks, moves cards, records hours	Assigned tasks; timesheet entry
Client	Follows progress and approves deliverables	Portal access to the project and its tasks
Controller	Checks hours, costs and invoicing	Profitability report; timesheet approval

2 The building blocks of the Project app

About a dozen objects explain the whole app. The table gives each one a plain meaning, an example and its technical name.

Object	What it is	Example	Technical name
Project	The container: name, customer, manager, privacy, stages	Alborz Switchgear Installation	`project.project`
Task	The unit of work: title, assignee, deadline, stage, hours	Install panel 3	`project.task`
Stage	A column of the board; can be shared by projects or made per project	New, In progress, Done	`project.task.type`
Milestone	A dated checkpoint on a project	Site handover	`project.milestone`
Subtask	A task with a parent task	Test panel 3	parent_id on a task
Dependency	A task that waits for another one	Counter waits for cupboards	`depend_on_ids`
Tag	A free label used for filtering and reports	Urgent, Client A, Rework	`project.tags`
Priority	Star rating of urgency	★★★	priority
Recurrence	A task that repeats on a rhythm	Weekly site inspection	recurrence
Timesheet line	One record of hours on a task by a person on a date	6.5 hours, Reza, 14 Azar	`account.analytic.line`
Template	A set of tasks you copy into a new project	Standard installation checklist	is_template
Followers	People notified of changes	Client, manager	message_follower_ids

Privacy and visibility

Setting	Who sees the project	Use it when
Invited employees	Only the people added to the project	Client work, anything private
All employees	Every internal user	Shared internal initiatives
Portal users	Also the clients invited to the portal	The client should follow progress
Followers only	Only the followers	Sensitive internal projects

> **Privacy is per project, not per task**
>
> A task inherits its project's privacy. If a task needs to be hidden from part of a team, the project is the wrong container; split the work into two projects. Changing a project from private to shared later changes what its tasks reveal, so decide before the first client is invited.

Stages: one board or many?

A stage can be shared by every project, or belong to a single project. Shared stages keep reports comparable across projects; project-specific stages suit projects with a very different process. Most teams start with shared stages and add specific ones only when the work demands it. A task moving stage can fire automatic actions, such as emailing the client when the task reaches Review.

3 Setting up Projects

The set-up in dependency order. Each step uses the one before it.

Step 1 Enable the features

Project › Configuration › Settings

Setting	What it switches on	Turn it on when
Tasks in project	Tasks inside a project, the kanban board	Almost always
Milestones	Dated checkpoints on a project	The client expects delivery dates
Subtasks	Tasks inside tasks	Work has clear sub-steps
Dependencies	Tasks that wait for others	Order matters; a delay should show up
Recurring tasks	Tasks that repeat on a schedule	Inspections, reports, maintenance
Timesheets	Recording hours on tasks	You bill or cost by hour
Profitability	Revenue and cost of the project in one view	You want margin visible
Project Sharing	Client portal access	Clients follow progress themselves
Gantt, Project Updates Enterprise	Timeline view; status reports sent to clients	Long projects with dependencies, or client reporting

Step 2 Stages and tags

Project › Configuration › Stages and Project › Configuration › Tags

- **Stages.** Keep four or five: New, In progress, Review, Done, and optionally Cancelled. Mark Done as the folded or closed stage so it stays out of the way.
- **Tags.** Use a small, agreed list: client, urgency, type of work. A tag nobody searches for is noise.

Step 3 Projects and templates

Project › Projects

Field	What it drives
Name and customer	What the client sees and what sales documents link to
Project manager	Who receives the project dashboard and approves changes
Privacy and visibility	Who sees the project (section 2)
Stages	Which columns the project uses

Field	What it drives
Analytic account	Links the project to accounting, so hours and costs get a home in the ledger
Default dates	Start and end dates used for the Gantt and planning

Use a template for work you repeat. A checklist of 20 standard tasks saves a manager an afternoon on each new client, and it keeps every project of the same type comparable.

Step 4 Timesheets and rates

Project › Configuration › Settings › Timesheets

Item	Where	Effect
Timesheet Rates	Employee and project	The cost per hour for each employee, used for profitability
Billing rate	Product (service) and project	The sale price per hour, used for invoicing
Timesheet approval	Settings	Hours must be validated by a manager before they count
Timer	Timesheet entry	Start and stop a timer from the task; the hours are saved on stop

Iran Iranian practice

- Hourly rates are often agreed in rials or in toman. Store the rate in rials; show toman on the screen only if the client asks.
- Fiscal year runs from 1 Farvardin. Projects that span two years need timesheets split by accounting date, which Odoo does by default.
- Use the Jalali calendar in your head and in the client conversation; Odoo dates stay Gregorian (see section 9).

Outside Iran Elsewhere

- Time-based billing with a rate card per role is common in consultancy; set the rates on the employee and the service product.
- Fixed-price projects use milestones with invoices per milestone, not per hour; the profitability view still shows the hours behind them.
- Multi-currency clients: the project currency follows the sales order.

4 Working in the views: kanban, list, calendar, Gantt

The same tasks, four ways of looking at them. Each view answers a different question.

View	Question it answers	Best for	Where
Kanban	What is each person doing, and where is each task?	Daily work; moving cards between stages	Project › Tasks › Kanban
List	Which tasks are late, and how many hours are left?	Filtering, sorting, bulk changes	Project › Tasks › List
Calendar	What is due this week?	Deadlines and personal planning	Project › Tasks › Calendar
Gantt Enterprise	When does each task start and finish, and what waits for what?	Long projects; dependencies; delays	Project › Tasks › Gantt
Pivot and graph	How many hours or tasks per person, stage or month?	Management reports	Project › Reporting

Reading the kanban board

Each card shows the title, the assignee's picture, the deadline, the priority stars and the tags. A red deadline means late. A card with a small paperclip has attachments; a card with a speech bubble has messages. Drag a card to move it; the stage change is saved and logged in the chatter.

Four filters that do most of the work

- **My tasks:** everything assigned to you, across projects.
- **Late:** tasks past their deadline and not done.
- **Unassigned:** work nobody owns yet; clear this every morning.
- **By milestone:** everything that must be ready for the next checkpoint.

The Gantt view Enterprise

The Gantt shows tasks as bars on a timeline, with dependencies as arrows. Drag a bar to move it; if a task it depends on moves, the dependent tasks move with it. The red line marks today. This is the view that shows a delay coming: when the counter bar is pushed past the handover milestone, the schedule has a problem, even if nobody has said so.

> **Gantt on a Jalali calendar**
> Odoo draws Gregorian months. A team working in Jalali months can read the Gantt with a printed Jalali calendar next to it; the dates in the database are unaffected. Make the milestone dates exact Gregorian dates so that the Jalali conversion does not shift a handover by a day.

5 Tasks, milestones and dependencies, in depth

The three things that turn a list of tasks into a plan: who does what, when it is due, and what must happen first.

A task, field by field

Field	What it drives
Title	What people search for; write it as an action: “Install panel 3”
Assignees	Who answers for it; several people can share a task
Deadline	Shown in red when late; used by the calendar and the late filter
Stage	Where the card sits; moving it can notify the client
Priority	The stars; used for sorting in the list view
Tags	For filtering and for reports
Milestone	The checkpoint the task supports
Parent task	Makes it a subtask; the parent shows progress of its children
Planned hours, allocated time	How much work was estimated; compared against recorded hours
Description, attachments	The instructions and the evidence

Milestones: the checkpoints that matter to the client

A milestone is a date the client cares about, such as “site ready for handover”. Attach the tasks that must be complete before it. A milestone can be marked reached only when those tasks are done, and the project dashboard shows how many milestones are on track. A project without milestones can run forever; a project with three clear milestones has three promises.

Milestone	Date	Tasks that must be done	Status
Cabinets in place	25 Azar 1405	Remove old units; fit wall units; fit base units	On track
Counter fitted	5 Dey 1405	Measure counter; cut; fit counter	Waiting
Site handover	15 Dey 1405	Electrics tested; cleaning; client walkthrough	Not started

Dependencies: what must wait for what

A task can depend on other tasks. While a predecessor is open, the dependent task is blocked and cannot be started, and in the Gantt its bar moves when the predecessor moves. Dependencies are the most useful way to see that a delay on one task will reach the handover, without anyone reading the whole plan.

1. **Finish to start** is the standard: the counter starts after the cupboards finish.
2. **Keep dependencies short.** A chain of thirty tasks is hard to read; group them under milestones.
3. **Do not make cycles.** If A waits for B and B waits for A, nothing can start. Odoo refuses the second link.

Recurring tasks

A recurring task creates the next occurrence when the current one is done, on the schedule you set: weekly, monthly, or every N days. Use it for inspections, reports and maintenance visits. Recurrence is the correct tool for repetition; copying tasks by hand creates duplicates that nobody closes.

> **Rule of thumb**
>
> A milestone answers “when do we deliver?”, a dependency answers “what must happen first?”, and a deadline answers “when should this task be finished?” Confusing the three is the most common planning mistake.

6 Connections: Sales, Timesheets, Accounting, HR

A project is useful alone; it becomes powerful when it is connected to what the company sells, what people cost, and what the books record.

From sale to cash, through the project

Sales order → Project created → Tasks and hours → Invoice (rate or milestone) → Payment

Connected app	What flows	How to set it up
Sales	A sales order with a service product creates a project automatically; milestones on the order create tasks	Service product with “Create a Project” set; invoicing policy by milestone or hours
Timesheets	Hours recorded on tasks become invoice lines at the billing rate, or cost in profitability	Timesheet rates on employees; billing rate on the service
Accounting	The project’s analytic account receives every timesheet, bill and invoice line tagged to it	Analytic account on the project; analytic distribution on bills and invoices (see the Budgets guide)
HR	Employee cost per hour, working schedule, leaves that reduce capacity	Employee timesheet cost; working hours; time off
Purchase and Inventory	Materials bought for a project can carry its analytic account	Analytic distribution on purchase orders
Helpdesk, Planning Enterprise	Support tickets turned into tasks; staff planned across projects	Helpdesk team linked to a project; Planning for capacity

Billing from timesheets: an example

Sahand Industrial bills Alborz Trading Co. for engineering support at a rate of 3 million IRR per hour. In November, two engineers record 40 and 25 hours on the project tasks. The invoice is created from the timesheets, not typed.

Line	Hours	Rate (million IRR)	Amount (million IRR)
Engineer A, Installation support	40	3.0	120
Engineer B, Testing support	25	3.0	75
Total before VAT	**65**		**195**
VAT 10%			19.5
Invoice total			**214.5**

The timesheet entries are the evidence: each line can be opened and traced to a task, a date and a person. If the client questions the invoice, the answer is in the project, not in an engineer's memory.

Hours that were not billed

Hours recorded on an internal task, or on a project billed at a fixed price, do not create invoice lines. They still count in profitability as cost. A project that looks profitable on invoices can be a loss once its unbilled hours are added, so check the cost side every month.

Iran Practical points

- Cost per hour for an engineer should include the employer's insurance share, which is part of the real cost.
- A Persian invoice may need the economic code and the tax ID of the client; these come from the customer record, not the project.
- Postdated cheques for project milestones: the invoice follows the milestone; the cheque is a payment matter (see the Accounting guides).

Outside Iran Practical points

- Multi-country projects: set the currency on the sales order; profitability converts to the company currency at the accounting date.
- VAT on services follows the place of supply; check the tax on the service product before the first invoice.
- Subcontracted hours from a partner firm are recorded as a vendor bill on the project, not as hours.

7 Strengths and weaknesses

An honest assessment of the Project app, so that a team knows where it fits and where it needs help.

Strengths

Strength	Why it matters
One place for work, hours and money	Tasks, timesheets, invoices and costs share one database; no export is needed to see a margin
Simple to start	Kanban and list views need almost no training; a new user moves a card in minutes
Good connection to Sales and Accounting	Sales orders create projects; timesheets create invoice lines; analytic accounts carry the costs
Client portal	Clients follow progress and comment without email chains
Templates and recurrence	Repeated work is set up once and reused
Flexible views	The same tasks can be read as a board, a list, a calendar, a timeline or a report
Dependencies and milestones	Delays show their effect on dates, not only on a status
Transparent history	Every stage change, assignment and message is logged in the chatter

Weaknesses

Weakness	What it means in practice	How teams cope
Not a full portfolio or resource tool	Comparing many projects against one team's capacity is limited without Planning Enterprise	Use Planning Enterprise or a spreadsheet for capacity; keep project lists clean
Gantt and some reports are Enterprise	Community users work with lists, kanban and pivot only	Plan the budget; use list and kanban as the working tool
Dependency rules are simple	Mostly finish-to-start; complex lead and lag patterns need care	Model the critical path manually for complex work
Easy to create too many projects	Hundreds of stale projects clutter the menus and the reports	Archive projects at closure; review the list monthly
Timesheets depend on discipline	Hours not entered are costs not seen; the margin is too good	Daily reminder; approval step; unbilled hours report
Privacy is per project	A task cannot be hidden on its own	Split the work into projects where confidentiality matters
Reports need set-up	The default dashboard answers basic questions only	Build a few pivot views and save them as favourites

Weakness	What it means in practice	How teams cope
Mobile use is limited in practice	Editing many tasks on a phone is slow	Use the mobile app for quick updates; the desktop for planning

A quick scorecard

Need	Fit	Comment
Small team, one or a few projects	Strong	Kanban and timesheets are enough
Client services and consultancy	Strong	Billing from timesheets; milestones; portal
Software or product delivery with sprints	Good	Works well for tasks; agile boards need configuration
Construction with many dependent trades	Partial	Gantt Enterprise helps; heavy scheduling may need a specialised tool
Project costing against a fixed budget	Good	Profitability; analytic accounts; budgets
Capacity planning across many people	Limited	Planning Enterprise is the right add-on
Full project portfolio governance	Not native	Use a portfolio tool or a spreadsheet for the board view

8 Real-world challenges and how Odoo solves them

Eight situations, each with the problem, the set-up that answers it and what to watch for. Well-known companies illustrate the problem only; they are not claimed to use Odoo.

1 Client projects with a fixed price: the Airbus problem

Large engineering work is delivered in phases, and each phase has a sign-off. Fictional case: Sahand Industrial, a switchgear contract.

Problem	A fixed-price contract of 2,400 million IRR in three phases. The team does not know whether it is still making money halfway through.
Odoo set-up	One project per contract; milestones per phase; invoicing by milestone on the sales order; timesheets on every task for cost.
What happens	Phase 1 is delivered and invoiced at its milestone; the profitability view shows the hours spent against the phase's share of the price.
Watch for	A fixed price with unlogged hours shows a false margin. Make timesheets mandatory on the phase tasks.

2 Hours on the wrong project: the consultancy problem

Engineers work on three clients a week and rarely remember which hours belong where. Fictional case: a services firm.

Problem	Invoices are wrong or disputed because hours were recorded late, rounded, or on the wrong client.
Odoo set-up	Timesheets entered daily on the task; the timer for live work; approval by the project manager each week; billing rate per employee and service.
What happens	The weekly approval shows each engineer's hours by task. Disputed lines are traced to a task, a date and a description.
Watch for	The timer can run through lunch; a daily check of hours above eight is a simple control.

3 A launch that slipped: the Toyota problem of flow

Work moves through stages, and a bottleneck in review slows everything. Fictional case: a software team shipping a product update.

Problem	Tasks pile up in Review; nobody knows which stage is the constraint.
Odoo set-up	Stages New, In progress, Review, Done; a count of tasks per stage on the dashboard; a weekly look at the Review column with its age.
What happens	The team sees that eleven tasks wait in Review for more than three days; reviewers are scheduled to clear the column first.
Watch for	Stage counts only work if people move cards. A board that is never updated tells a false story.

4 Dependencies that hide a delay: the construction problem

A late delivery of a part stops three trades. Fictional case: a fit-out project in Tehran.

Problem	The delivery of the switchboard slips by a week, and the painters, electricians and installers each find out late.
Odoo set-up	Tasks linked by dependencies; the Gantt view Enterprise or the dependency list; milestones on the handover date.
What happens	Moving the switchboard task shows the dependent tasks and the handover milestone at risk in the same screen.
Watch for	Dependencies must be kept up to date. An unlinked task is invisible to the chain.

5 Repeated inspections: the maintenance problem

A plant must inspect equipment every month and keep proof. Fictional case: a food processing plant.

Problem	Inspections are forgotten, or recorded in paper books that nobody can search.
Odoo set-up	A recurring task monthly per machine; a checklist in the description; attachments of the signed report; a late filter.
What happens	The next inspection appears when the last one is done; the late list shows any skipped month; the attachment is the evidence.
Watch for	A recurring task that nobody closes piles up. Assign an owner for each machine.

6 Client visibility without status calls: the agency problem

A marketing agency sends weekly updates by email that take a day to write. Fictional case: a digital agency.

Problem	Clients ask for progress constantly; the account manager spends hours on status emails.
Odoo set-up	Project Sharing for the client; the project visible to portal users; milestones that the client sees; comments on tasks.
What happens	The client opens the project and sees the milestones, the tasks in progress and the comments; the weekly email is a summary link.
Watch for	Clients see what is shared. Keep internal notes on internal tasks, not on client-visible ones.

7 Projects that cost more than they bring: the margin problem

A project looks profitable until its unbilled hours are counted. Fictional case: Sahand Trading, a supply project in Dubai.

Problem	The margin on the invoices is 25%; the real margin, with internal hours, is negative.
Odoo set-up	Timesheet cost per employee; profitability per project; the unbilled hours report; a monthly review of projects below a margin threshold.
What happens	The profitability view shows cost and revenue side by side; the project is re-scoped before the next phase.
Watch for	Cost per hour must be realistic; a rate that ignores overheads flatters every project.

8 Tracking many small projects: the internal initiatives problem

A company runs twenty internal improvement projects with different owners. Fictional case: Sahand Industrial operations.

Problem	Nobody knows which initiatives are active, which are late, or who owns what.
Odoo set-up	One project per initiative with a manager; shared stages; tags for the type of initiative; the pivot report by manager and stage.
What happens	The monthly review uses one pivot: initiatives per stage and owner, with the late ones first.
Watch for	Projects without an owner become orphans; a mandatory manager field stops that.

9 Iran and the rest of the world on one page

Project management is the same discipline everywhere. What changes is the calendar, the currency, the tax and the words people use.

Topic	Iran	UAE, our example	Elsewhere, for comparison
Calendar	Jalali year and months; Odoo stores Gregorian dates; show Jalali dates in the client conversation and on printed plans	Gregorian	Gregorian
Billing	Rials; invoices carry the economic code and the client tax ID; milestones often paid by postdated cheque	Dirham; VAT 5%; bank transfer	Local currency; VAT or sales tax by country
Rates	Hourly rates agreed in rials; inflation pushes rates up each year	Rate cards per role	Rate cards, fixed price, or retainers
Vocabulary	Project پروژه, task وظیفه, milestone نقطه عطف, timesheet برگه زمان, budget owner مسئول پروژه	Project, task, timesheet	Project, task, timesheet, sprint
Hours	Working week and holidays follow Iranian public holidays; set them in the working schedule	Working week follows local rules	Local calendars
Language	Persian task titles are fine; keep one language per project for the reports	English and Arabic	English

What standard Odoo does not do for Iran

Jalali dates on screens and prints, Jalali Gantt labels, Persian reports and print layouts, and a link to tax or e-invoicing systems. Nothing in this guide changes: tasks, hours and money work the same way in every country.

Iran A sensible Iranian set-up

- One project per client contract; milestones matching the payment schedule.
- Rates in rials, with a review each Farvardin.
- Working schedule set with the Iranian holidays.
- Milestone dates written both in Jalali and Gregorian in the description.

Outside Iran A sensible UAE set-up

- One project per client, with analytic account for profitability.
- Rate cards per role, reviewed yearly.
- Milestone invoicing for fixed-price work; timesheet billing for the rest.
- Working schedule with the local weekend.

10 A 50-task deployment walkthrough, explained simply

How a team deploys Odoo for a client, from the first call to the first month of live use. Each task is written so that a new team member can follow it one by one. The Project app is the place where all 50 are tracked.

How to read this table

Each row is one task. **Type** says what kind of work it is: **Analytical** (understanding the business), **Functional** (configuring Odoo through its screens), **Technical** (servers, integrations, data loading, code), **Management** (planning, meetings, risk, reporting) or **Training and go-live**. **Owner** is the role that answers for it. **Output** is what must exist when the task is done. The client used in the example is Sahand Industrial, with Sahand Trading as its sister company.

Phase A: Start and understand (tasks 1 to 10)

#	Task	Type	Owner	Output
1	Kick-off meeting with the sponsor; agree goal and scope	Management	Project manager	Kick-off minutes
2	Name the client team and their roles	Management	Project manager	Contact list with roles
3	Write the project charter: goals, scope, out-of-scope items, dates	Management	Project manager	Charter approved
4	Collect the current systems and spreadsheets in use	Analytical	Functional consultant	Inventory of current tools
5	Map the main business processes: sales, purchases, stock, finance	Analytical	Functional consultant	Process maps
6	Interview the department heads, one by one	Analytical	Functional consultant	Interview notes
7	List pain points and quantify the time or money lost	Analytical	Functional consultant	Pain-point list with numbers
8	Confirm the country set-up: currency, tax, calendar, language	Analytical	Functional consultant	Country set-up sheet
9	Agree which Odoo apps are in scope for the first release	Analytical	Functional consultant	Scope list by app
10	Set up the project itself in Odoo: stages, milestones, owners	Management	Project manager	Project with all 50 tasks

Phase B: Design the solution (tasks 11 to 20)

#	Task	Type	Owner	Output
11	Design the company structure: companies, branches, currencies	Functional	Functional consultant	Company design
12	Design the chart of accounts and the groups (Group, Kol, Moein, Tafsil)	Functional	Accountant	Chart of accounts
13	Design the tax set-up: taxes, tax groups, return boxes	Functional	Accountant	Tax design
14	Design the journals and numbering sequences	Functional	Accountant	Journal and numbering list
15	Design the sales process: quotation to invoice to delivery	Functional	Functional consultant	Sales process design
16	Design the purchase process: request to receipt to bill	Functional	Functional consultant	Purchase process design
17	Design the warehouse: locations, steps, routes, lots	Functional	Functional consultant	Warehouse design
18	Design manufacturing: BoMs, work centres, quality points	Functional	Functional consultant	Manufacturing design
19	Identify the integrations: bank, e-invoicing, website, payroll	Technical	Technical lead	Integration list with effort
20	Review the design with the client; get written sign-off	Management	Project manager	Signed design document

Phase C: Build and load data (tasks 21 to 30)

#	Task	Type	Owner	Output
21	Create the test database with the chosen apps and country set-up	Technical	Technical lead	Test database
22	Configure companies, currencies and fiscal year	Functional	Functional consultant	Company set-up done
23	Configure the chart of accounts and default accounts	Functional	Accountant	Chart loaded
24	Configure taxes and fiscal positions	Functional	Accountant	Taxes tested
25	Configure sales, purchase and warehouse flows	Functional	Functional consultant	Flows tested in test database
26	Clean the master data: customers, vendors, products	Technical	Data team	Clean master data file
27	Write the import templates and load master data in test	Technical	Technical lead	Master data loaded
28	Load opening balances in test; reconcile to the old system	Technical	Accountant	Opening balance reconciliation

#	Task	Type	Owner	Output
29	Configure reports, dashboards and the project views	Functional	Functional consultant	Reports list
30	Build any custom piece agreed in the design, with its own tests	Technical	Developer	Custom module tested

Phase D: Test (tasks 31 to 40)

#	Task	Type	Owner	Output
31	Write the test scenarios from the process maps	Functional	Functional consultant	Test scripts
32	Run the unit tests: each app by itself	Functional	Functional consultant	Unit test results
33	Run end-to-end tests: quote to delivery to invoice to payment	Functional	Functional consultant	End-to-end results
34	Test the integrations: bank, e-invoicing, website	Technical	Technical lead	Integration test results
35	Test security: who can see and change what	Technical	Technical lead	Access matrix tested
36	Test performance with realistic data volumes	Technical	Technical lead	Performance report
37	Run user acceptance tests with client staff	Management	Project manager	UAT sign-off
38	Log every defect; prioritise; fix the critical ones	Management	Project manager	Defect log, closed
39	Re-run the tests after fixes; confirm no regressions	Functional	Functional consultant	Clean test run
40	Test the closing of a month and a year in the test database	Functional	Accountant	Closing test passed

Phase E: Train, cut over and go live (tasks 41 to 47)

#	Task	Type	Owner	Output
41	Write the user guides for each role, in the client's language	Training and go-live	Functional consultant	Role guides
42	Train the super-users first; then the users, by role	Training and go-live	Functional consultant	Trained users
43	Agree the cut-over date and the freeze of the old system	Management	Project manager	Cut-over plan
44	Run the final data load and the final opening balance	Technical	Technical lead	Production loaded
45	Reconcile production to the old system; sign it off	Analytical	Accountant	Reconciled production

#	Task	Type	Owner	Output
46	Switch the users to Odoo; watch the first day and the first week	Training and go-live	Project manager	Go-live log
47	Hold the daily stand-up for two weeks; log every question	Management	Project manager	Issue list, closed

Phase F: Support and close (tasks 48 to 50)

#	Task	Type	Owner	Output
48	Set up the support model: who answers, in what time, for what	Management	Project manager	Support agreement
49	Measure the results against the pain points of task 7	Analytical	Functional consultant	Before and after report
50	Hand over the documentation and close the project in Odoo	Management	Project manager	Handover pack; project archived

Reading the walkthrough as a plan

The 50 tasks fall into six phases, and the phases overlap. Testing begins while the last configuration is still being done; training starts once the test has passed the end-to-end scenarios. Put the phases in the project as milestones, and put the tasks above in the phase they belong to. Dependencies are the real plan: task 27 cannot start before task 26, and task 45 cannot be signed before task 28 is reconciled.

> **Where each task lives in the Project app**
>
> Create one project named after the client, one stage per phase or the shared stages, one milestone at the end of each phase (design sign-off, test sign-off, go-live, handover), and the 50 tasks with owners and deadlines. Record hours on each task. Each month, the profitability view will then show whether the deployment is on budget.

11 Profitability and reporting

The project is the place where cost, revenue and time meet. This section explains the reports that show whether the project is healthy.

Report	Shows	Question it answers
Project dashboard	Tasks by stage, late tasks, milestones, hours	Where does the project stand this week?
Profitability	Revenue, costs, margin per project	Is this project making money?
Timesheets report	Hours by person, task, project and week	Who spent the time, and on what?
Unbilled hours	Hours recorded but not yet invoiced	What have we done but not charged for?
Pivot by stage and manager	Tasks per stage and owner	Where is the bottleneck?
Gantt Enterprise	Tasks on a timeline with dependencies	Which dates are at risk?

A profitability example

Sahand Trading runs a supply project for Gulf Lighting LLC. The sale is 200,000 AED. Costs are timesheets at internal rates plus purchased goods. Amounts in AED.

Line	Budget	Actual	% of budget
Revenue (sale price)	200,000	200,000	100.0%
Timesheet cost (engineers, 320 hours)	−60,000	−72,000	120.0%
Purchased goods	−110,000	−104,000	94.5%
Subcontractor	−10,000	−16,000	160.0%
Margin	**20,000**	**8,000**	**40.0%**

Reading the table: the margin is 8,000 instead of 20,000. The timesheet and subcontractor lines are over budget; the goods line is under. The project manager now knows where to act: check whether the extra hours were scope creep, and whether the subcontractor should be replaced for the next phase.

> **Profitability is only as good as the hours**
> If engineers do not record their hours, the cost line is too low and the margin too high. Look at the number of hours per week per person each month, and compare it to what the calendar says they were working. A gap is the first signal of a problem.

12 Go-live checks and common mistakes

Run these in a copy of the configured database before the project is used with real clients. Each test exercises one link in the chain.

Step 8 Twelve tests

#	Do this	Expect this
1	Create a project from a confirmed sales order with a service product	The project appears, linked to the order and the customer
2	Add a milestone and link two tasks to it	The milestone shows the tasks and their status
3	Add a dependency between two tasks; move the first task's deadline	The dependent task shows the change, in the Gantt if available
4	Record 6.5 hours on a task with the timer, then stop it	A timesheet line of about 6.5 hours is created for the employee on the task
5	Approve the week's timesheets as the manager	Approved hours are visible; unapproved ones are flagged
6	Invoice the project from the timesheets at the billing rate	Invoice lines match the hours and the rate; the hours are marked as invoiced
7	Look at the profitability of the project	Revenue, cost and margin match the invoice and the timesheet cost
8	Check that the project analytic account receives the bills and invoices	The analytic items appear under the project account
9	Invite a client to the portal; log in as the client	The client sees only their project and the shared tasks
10	Make a task private inside a private project; log in as a team member not invited	That member cannot see the project or the task
11	Create a recurring task monthly; complete one occurrence	The next occurrence is created with the next date
12	Archive a finished project	It leaves the active lists but its timesheets and invoices remain in reports

Common mistakes

Mistake	What you see	Remedy
No deadlines on tasks	The late filter is empty; nobody is late; nothing is	Every task gets a deadline on creation
Tasks without an owner	Unassigned work piles up	Clear the unassigned filter daily

Mistake	What you see	Remedy
Hours not recorded	Margin looks too good	Daily reminder; weekly approval; unbilled report
Hundreds of stale projects	Menus slow; reports noisy	Archive on closure; monthly review
Milestones used as tasks	Milestones never reached; reports confused	Milestones are dates; tasks are work
Stages renamed or deleted in use	Tasks move unexpectedly	Rename safely; plan deletions
Client sees internal notes	Embarrassment; confidentiality breach	Internal comments on internal tasks; check the portal before sharing
Dependencies missing	A delay is not seen until late	Link every task that waits for another
Billing typed by hand	Invoices disagree with hours	Invoice from timesheets
No analytic account on the project	Costs of the project are not in the ledger	Set the analytic account on creation
Hours on the wrong task	Two projects look wrong at once	Approval by the manager each week

13 Cheat sheet, glossary and self-check

Where each setting is, what it is called in Persian, and eight questions to test yourself.

Where is it?

Step	Setting	Menu	Object or field
1	Features	Project › Configuration › Settings	Tasks, milestones, timesheets
2	Stages	Project › Configuration › Stages	`project.task.type`
2	Tags	Project › Configuration › Tags	`project.tags`
3	Projects	Project › Projects	`project.project`
4	Timesheet rates	Employee › Timesheet cost	Hourly cost per employee
5	Tasks, milestones	Project › Tasks	`project.task` , `project.milestone`
6	Timesheets	Timesheets › My / All	`account.analytic.line`
6	Profitability	Project › Reporting › Profitability	Revenue, cost, margin
7	Portal	Project › Project › Share	Client access

Menu positions can differ slightly between 19.x releases. If a path does not match, use the search at the top of the screen with the name in the second column.

Persian, English, Odoo

Persian	In Odoo
پروژه	Project
وظیفه	Task
نقطه عطف	Milestone
زیروظیفه	Subtask
وابستگی	Dependency
مرحله	Stage

Persian	In Odoo
برگه زمان	Timesheet
نرخ ساعتی	Hourly rate (billing rate or cost rate)
سودآوری	Profitability
ساعت ثبت‌نشده	Unbilled hours
تکرارشونده	Recurring task
قالب	Template

Check yourself

	Question	Answer
1	What is the difference between a milestone and a dependency?	A milestone is a dated checkpoint the client cares about; a dependency says that one task must wait for another.
2	Why must every task have a deadline?	So that the late filter and the calendar work, and so that delays are visible.
3	Where are the hours of a task stored?	As timesheet lines on the task, linked to the employee and the analytic account.
4	A client asks for a status update. What do you offer?	The portal, where the client sees the milestones and the tasks shared with them.
5	A project shows a good margin, but nobody records hours. Why is that a problem?	Cost is missing, so the margin is overstated; the real margin is lower.
6	How do you bill a client from timesheets?	Create the invoice from the timesheets; the lines are generated at the billing rate.
7	Can a single task be made private inside a shared project?	No. Privacy is per project; split the work into two projects if needed.
8	What should be done before the first client is invited to the portal?	Check privacy and visibility, remove internal notes from client-visible tasks, and confirm the stages.

The whole guide in two lines

A project is a box of tasks with owners, stages, milestones and dependencies; the hours recorded on its tasks become the invoices and the margin. Get the tasks right, record the hours, and the project tells the truth.