

Coderbunker & clients collaboration

Why do clients choose us?

- fast turnaround finding suitable resources
- product management and architecture leadership
- depth and breadth of technical expertise
- iterative and agile development enable fast pivots
- manageable risk

Why do talented freelancers join us?

- other talented and passionate people to work with
- more interesting and diversified projects
- better pay (less overhead)
- easy to join (no complicated hiring process)
- senior experience from members like Ricky and Fred
- enjoyable diverse multifaceted community
- flexible schedule
- self-determined compensation
- minimal bureaucracy
- confidence that the work done will be fully compensated

What makes us different

We work collaboratively to be an extension of your team, to have the same goals as the team.

- Offer flexibility to our clients constantly changing business needs
- Be supportive of our community freelancers across all our projects
- We focus on simplicity through
 - radical transparency (post aggressively in group chats, share documents, real-time tracking)
 - minimal administrative overhead (prepaid retainer)
 - consistent development practices (automated testing, continuous integration, code reviews)
 - common technology stack (React, NodeJS, PostgreSQL, Python, ...)
 - budget optimization (balance between junior and senior resources)
 - empowerment of engineers (initiatives, shared access to resources, collective ownership)
- An hour worked is an hour paid
 - including administrative and setup overhead
- We don't do fixed projects
 - Instead we fix budget and fix goals
- We don't build in multipliers for risk
 - you pay for resources, not deliverables
- You only pay for the engineering and non-engineering resources you need
 - Technical leadership, project and product management, etc are all options

Organizational innovation

Due to the unique set of rules Coderbunker applied consistently across our customers, we've noticed significant improvements in the relationship between clients and freelancers compared to traditional employee, outsourcing or freelancing relationships:

- Flexibility leads to more stable enthusiastic relationships
 - A number of projects that have had the same freelancers participating for over a year
- Transparency leads to smooth transition
 - When encountering a blocking issue, they are more likely to find a way out
- Collaborative learning leads to a standardized set of technologies and practices
 - Since we share our learnings to the community, we're more likely to be able to reuse these practices across projects and onboard people with prior knowledge
- Direct relationships between clients and freelancers improve productivity
 - Senior leadership does not mediate between clients and freelancers - we intervene on clients or junior freelancers requests
- Hour-by-hour tracking encourages optimal use of resources
 - By reflecting real cost in real-time, we can quickly make adjustments instead of pursuing solutions that are too expensive
- Resources as the deliverable
 - Enables quick pivots of the product
 - Adaptable work
 - Lesser costs by not doing rigid upfront requirements
- Focus on simplicity lessen conflicts
 - Fewer hardships in the relationship between developers and employers
 - Lower communication overhead

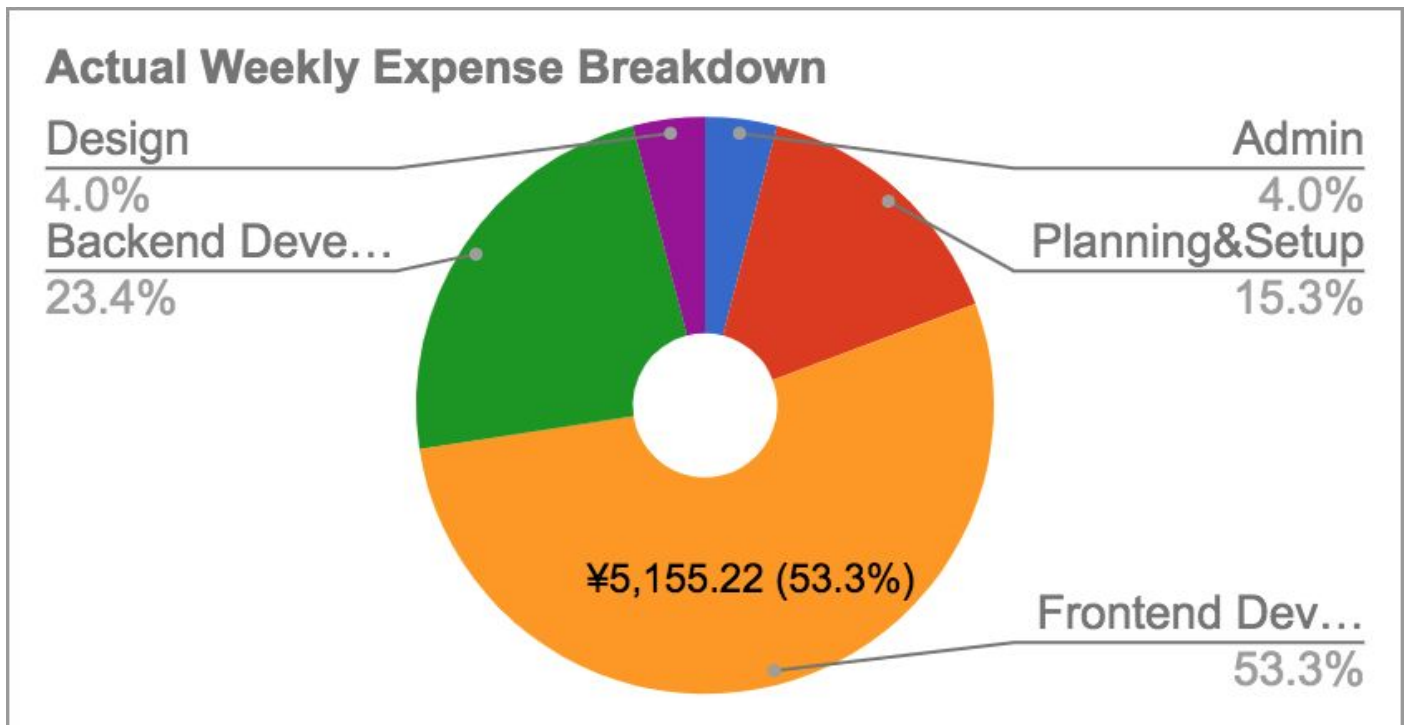
Collaboration mechanisms

- Senior developers should be kept in the day to day loop of all development projects
 - **Why?** Act as mentors/coaches/advisers across all projects. Can also suggest additional resources from the wide community on a as needed basis. Can advise individual developers and enforce consistent development practices and philosophy.
 - Senior developers involvement is generally low as a % of the budget and sporadic, generally driven by requests from clients and junior developers. By being supported by multiple projects budgets, senior developers are incentivized to stay involved in the community and enables them to make a living, offering their expertise at a relatively low cost for each individual client.
- Simplified approval process for resources
 - Junior developers need to be able (and must be encouraged to take the initiative) to reach out to other community members as necessary.
 - Founder and co-founder can also suggest good resources fit for approval to the client based on their current needs.
 - Allows to also try new resources and exit quickly those that don't fit.
- Communicate consistently technical decisions and actions in "client x Coderbunker" developer channels (WeChat, Github Issues, Github Projects...)
 - Removes the need for expensive weekly status update or progress update.
 - Allows senior developers to follow along and contribute suggestions (for resources or technology) as necessary and also update best practices across all projects.
- Every team includes founder and co-founders (Ricky, Fred) and legal entity representative (Teresa)
 - Ensures we can intervene quickly and support our members, coaching them offline to maximize performance and route new resources to the appropriate project.
 - Provide transparency to the administrative or non-value creating tasks overhead

Collaboration mechanics with the client

- Every project has (full client setup document is [documented](#)):
 - A Google Share folder specific for the client
 - Google Groups prefixed with client name
 - engineering team
 - our engineering team
 - administrative team for this client
 - Generally a Github organizational team (organization owned by client)
 - A timesheet
- We send a monthly (or in some case quarterly) invoice with sufficient time to account for transfer or payment terms

Example expense breakdown (project #1)



As presented, a pie chart (by our client Atlas AI Tech) of average weekly expense breakdown in another project that has run for 5 months.

This example project is highly technical (car telemetry data captured on mobile and streamed to the cloud) where the client and main stakeholder is able to do coding but has no large scale software development experience:

1. 4-5% admin costs specific are reflected by having our admin record hours in the project, administrative tasks, licenses (example: obtaining ICP), discussing organization
2. 15% planning & setup comes from developer recruitment & integration, environment setup, server setup, planning work

In a project where the client offers more experienced technical leadership from their side (tech lead) and/or have already established practices, the planning and setup percentage is possibly less.

Example resource breakdown (project #1)

Resource-wise the breakdown for this example project is as follows (additionally, a majority of the data processing algorithm was worked on by the client):

Resource	Rate/hour (RMB)	%
Junior React-Native Mobile Frontend Dev1	¥250.00	21.03%
Junior React-Native Mobile Frontend Dev2	¥250.00	17.22%
Senior Physics / Data processing Algorithm Dev	¥700.00	16.14%
Ricky (including development work on computer vision)	¥700.00	15.27%
Intermediate backend developer	¥350.00	5.75%
DevOps	¥350.00	5.37%
Embedded hardware developer	¥600.00	4.75%
Intern (development work on computer vision)	¥150.00	4.38%
Graphic designer / UI/UX design	¥150.00	2.61%
Junior Python Dev	¥250.00	2.19%
Tech administrative tasks (ICP, website, domain name)	¥150.00	2.07%
Fred	¥700.00	1.40%
Python Dev (trial basis, rejected)	¥250.00	1.20%
iOS Dev (trial basis, rejected)	¥330.00	0.60%
administrative	¥90.00	0.03%