

BRASSDECK

How Brassdeck works

The manual, and the connector reference

brassdeck.com/docs

How Brassdeck works

The first half is the app: what a card holds, what the six stages mean, and the one number that only moves when something real happened. The second half is the connector, which is how Claude or ChatGPT reads and works on the same board you do.

PART ONE, THE APP

What it is

Brassdeck holds everything you have started on one canvas, one card per idea, and keeps an honest record of where each one actually stands. It is not a list of things you owe: a card is an intention worth keeping, and stopping one on purpose is an outcome the app is built to make easy.

There is no ranking anywhere in it, no red, and no count of what you have failed to do.

The words

Eight words, each meaning one thing. The app uses them the same way everywhere, and so does the assistant when you connect one, because the connector is handed this list before it does anything.

WORD	MEANS
card	One idea. The thing you name, stage, star and write steps on.
canvas	The surface cards sit on, and the thing you pan and zoom.
board	All your cards together. What a share link covers by default.
deck	Your board as a list, which is what a phone gets.
view	A named subset of cards. A lens you choose.
step	One line written inside a card.
stage	Where a card is on spark, shaping, building, live.
workspace	Who is in it, what they can do, the billing boundary.

Two of them are worth pinning down. The canvas is the surface and the board is the content: you pan the canvas, and you share the board. And a stage says how far an idea has travelled toward being real, which is a different question from whether you touched it today.

The six stages

Four of them are a journey. The last two are exits, and they are not failures: they are kept, so the decision survives the idea.

01 Spark A thought, written down so it stops taking up room in your head.	02 Shaping Working out what it actually is, before building anything.
03 Building Actively being made.	04 Live It exists in the world.
exit Parked Deliberately stopped, with a date it comes back.	exit Dropped Decided against. Kept so the decision survives.

Parked and dropped are left out of every count in the app. Counting them would quietly turn your own decisions back into obligations.

The honest clock

Every card carries one number: how long since it last actually moved. Most tools lift a card to the top the moment you rename it, reorder it or drag it somewhere neater, so the list flatters you and the number stops meaning anything. This one is kept separately and moves only when something real happened.

A card dragged across the canvas after 147 days still reads 147 days. That is the whole point of it, and this is the full list rather than a rule of thumb.

WHAT YOU DID	MOVES THE CLOCK
You made the card	yes

WHAT YOU DID	MOVES THE CLOCK	
You set the next step	yes	Often the only thing between a card and moving.
You finished a step	yes	
You changed the stage	yes	
You parked it, or dropped it	yes	Stopping on purpose is an outcome, never neglect.
You added a step	no	Planning is not progress. A long list of intentions is not activity.
You renamed it, or edited its details	no	Renaming something you have not touched in a year must not reset the clock.
You moved the card on the canvas	no	Tidying your board is not progress.
You resized the card	no	
You starred it	no	A star says you are on it, which is not the same as having worked on it.
You looked at it and said it is still on	no	Looking at something is not working on it.
You skipped a check-in	no	

Parking or dropping counting as progress is not a trick. Treating a deliberate stop as neglect would be the same as shaming it.

Fading, not nagging

A card nobody has moved in a while recedes. Its paper tints toward the desk, the way paper left out does, rather than turning a colour meant to alarm you. The writing on it stays as readable as on any other card. The information is identical and the feeling is not, which is the entire argument.

- **The fade is scaled to each card's own cadence**, not to a fixed number of days. A quarterly idea untouched for forty days is fine. A weekly one untouched for forty days is not, and only the fade knows the difference.
- **A cadence of zero means never**. Set it and that card never fades, however long it sits.
- **Parked, dropped and finished cards never go cold at all**. They already stopped, on purpose. Counting the days since you last touched something because you finished it would be the meanest arithmetic this app could do.

Waiting on someone

Open a card and under the four stage buttons is a line that asks who or what you are waiting on. Write a name, a company, a queue. The card then says **Waiting on Wise support · 11d** on its face, and the day count is the half that matters: a label tells you what you already knew when you set it, and a number is a reason to go and chase somebody.

Waiting is not a stage, on purpose. A card waiting on a bank is still at building, and the day the bank answers it should still be at building rather than having to guess where it was. So the stage still says how far the idea got and the wait says who is holding it, and you never lose one to record the other.

A waiting card is no longer called stalled, because stalled means nobody has decided what happens next and here somebody has: they reply. The clock keeps running, though, and that is deliberate. A wait nobody has chased in six weeks is the single most useful thing this app can point at.

Neither starting nor ending a wait counts as progress. Marking it is not work, and the reply landing was somebody else doing it.

The star

A star means "I am on this right now". It is one bit. There is no second level, no ranking behind it, and starring a card is not progress: it does not touch the clock.

Star anything and a **Starred** row appears in the views menu, showing only those cards. It is built in rather than a view you made, so there is nothing to name and nothing to keep in step. Parking or dropping a card clears its star, because being on something you have deliberately stopped is a contradiction.

The room

Open a card and the panel offers **Mark it done**. Finished is not a seventh stage, deliberately: how far an idea travelled and whether you are done with it are two different questions, and they genuinely disagree. A thing can be live and still being worked on. A thing can be finished having never left building.

- **Where it goes.** A finished card leaves the working canvas for the room, a fixed place further down, laid out on a grid in the order things were finished.
- **How to get there.** The views menu, beside Tidy. It is a place rather than a filter, so you travel to it.
- **How to get back.** Escape, or the browser's Back button. The room pushes a history entry so Back does the obvious thing.
- **What it looks like.** Faded brass, never grey. A finished card keeps its accent and loses saturation. Draining the colour would file it with the cold ones, and it is not cold. Its meta line shows the date it was finished, where the clock used to be.
- **Reopening.** Nothing is deleted. Reopening puts the card back where it was on the canvas.

Finishing a card is for members of the workspace. Someone holding an editing link can tick steps and write, but cannot make a card vanish from a link somebody else is reading.

Views

A view is a lens, not a folder. A card can be in several of them and it never moves to get there. Switching is instant because everything is already on the page.

- **All cards** is the default, and Starred appears once you star something. Views you make sit under those, each with its count.
- **Tidy the canvas** lives in the same menu. It rewrites the position of every card you own, and the toast that follows offers "Put them back".

Three named pages ask a different question of the same board: **Focus** for how the load divides right now and what is actually moving, **Patterns** for why things stop, counted across every stall, and **Everything** for the whole board as one list ranked by how long each thing has sat.

Sharing

A share link is a bearer credential: whoever holds it has the access it was made with, and there is no sign-in behind it. Everything below is chosen when you make one, and can be changed on a link already handed out.

- **Viewer or editor.** A viewer can open any card in the link and read all of it, steps and history included, and change nothing. An editor can tick steps, write them and change what a card says. Someone opening an editing link is asked for a name and a colour, so you can see who changed what. That name is whatever they typed and is never verified.
- **Scope.** A link covers the whole board, or one view, or a single card. A view-scoped link shows only the cards in that view, and it says so on the link itself rather than looking like the whole board.
- **Expiry.** Never, unless you set one; a week and a month are there for a link you want to stop on its own. A link that has expired or been turned off shows the person a page saying so, with a way to ask you for a fresh one.
- **Two things are held back unless you turn them on.** Both default to off: **Stall reasons**, the reason you gave when something stopped, and **Dropped cards**. They are the most personal things in this app and they do not travel by default.
- **Revoking is permanent.** A revoked link is never revived, and it stays on the page with its view count, because how often something was opened is part of the record. Only a hash of the link is stored, so nobody, including whoever runs the app, can read a live link back out.

Ask

Ask is the assistant inside the app, in the toolbar and on the phone's tab bar. It runs the same tools the connector does, against your board, and it streams its answer as it works so you can see which tools it called.

You can also talk to it from inside a card. A line that begins with `@chit` in the step box or in the writing is a message about that card: "@chit break this into five steps" writes the steps, "@chit which mic under \$100" answers, and the answer is kept under your line in the

writing as "Chit:".

This is the opposite direction from the connector. Ask runs a model on your behalf and is billed per message. Connecting Claude or ChatGPT is not.

Routines

A routine is a standing notification: when something on the board reaches a number you chose, the workspace gets an email. You set one up by asking for it in plain words rather than filling in a form, and every routine fires **once** and then stops. A notification that arrives every night is a feed, and a feed gets muted.

KIND	FIRES WHEN
video_views	A YouTube link on a card passes a view count you named.
domain_free	A domain written on a card becomes available. The registry saying nothing is not the same as available, and is never reported as it.
project_quiet	A card goes untouched for a number of days. Parked and dropped cards are excluded, because they stopped on purpose.

Every routine is listed under Connectors with the number it is watching for and a way to stop it, whoever or whatever created it. The email quotes your own request back at you and carries nothing else.

What a step knows

Write a step that is nothing but a link or a name, and the app asks the world about it. There is nothing to connect and no key to paste. The rule is the same for all three: the whole line has to be the thing, because "look at that repo later" is a step and putting a reading on it would be answering a question nobody asked.

- **Domains.** A line that is a domain name shows whether it is still available. A label after a dash or a colon is fine.
- **YouTube.** A video link shows its views, and whether they have moved since the last look. A nightly snapshot is what turns a number into a direction. This one needs a key held by whoever runs the app, so it can be switched off; the Connectors page says which.

- **GitHub.** A repository shows how long since anyone pushed, which is the only number here that gets worse on its own if you do nothing. And a push moves the card: each night, a repository that was pushed to since the night before counts as real progress on every card that links it, so a card whose work is a repository is never called stalled while the commits are landing. A pull request or an issue shows what became of it, and when the pull request merges or the issue closes, the step that linked it is ticked for you. Parked and dropped cards are left alone.
- **Google.** A Docs, Sheets or Slides link reads as what it is. Titles and last-edited need a Drive connection, which is not built.

Three card sizes

Every card is the same width and one of three heights: **small**, **medium** or **tall**. Drag the bar on a card's bottom edge and it moves between the three. There is nothing in between, and that is the feature rather than a limitation.

The alternative is a card whose size is set by whatever happens to be written inside it, so adding a sixth step makes it taller than the card beside it and no amount of tidying will ever line the two up again. Three sizes means a canvas you can actually arrange, and it puts the decision back with you instead of with the content.

A card holding more than it can show clips and fades at the bottom edge rather than stretching. Nothing is hidden from you dishonestly: the count in the bottom corner has always been the real number of steps and still is, so a card showing three of them still says **4/11**.

All three heights are multiples of the canvas grid, so cards line up with each other and with the grid at the same time.

On a phone

A phone gets the deck rather than a shrunken canvas. An infinite canvas is a desktop idea: fit a real board onto a 375px screen and you land at twenty per cent, where a card is fifty-six pixels wide.

- **Deck** is the same cards and the same editor as a readable stack, ranked so the things asking something of you sit at the top.
- **Board** is one tap away and pinches to zoom properly.
- The raised round button in the middle makes a new card. It is the only control on the bar that makes something rather than showing something.

- **More** holds Focus, Review, Everything, Patterns, Share, Connectors and Settings.
- **Install it.** Add to home screen and it opens without browser chrome. On iPhone that is the share sheet, then Add to Home Screen.

Keyboard

On the canvas. All of these are ignored while you are typing in a field, except find and Escape, which belong to whatever is open.

N	A new card, on the canvas where you are looking.
/	Find. It travels to a match rather than filtering to it.
Cmd F	Find, and it works while the find field already has focus.
Cmd K	Ask, the assistant that runs on your own board.
=	Zoom in, about the pointer.
-	Zoom out.
0	Fit everything on screen.
Esc	Close what is open: find first, then the card panel, then the room.

While find is open

Enter	Travel to the next match.
Shift Enter	The previous one.
Down Up	The same, without leaving the field.
Esc	Close find and stay where you are.

Cmd is Ctrl on Windows and Linux. Opening a card gives that panel Escape, so it closes before the room does.

How it connects

Brassdeck speaks the Model Context Protocol, and it is the **server**. Claude and ChatGPT connect to it. The instinct is the other way round, and which way it goes is the whole reason this is free:

- No API key is needed from you, and nothing is billed here.
- The model, the agent loop and the conversation all stay in the assistant you already pay for.
- One endpoint serves Claude, ChatGPT and anything else that speaks the protocol, rather than two integrations that drift apart.

It is JSON-RPC 2.0 over a single POST to a URL that carries your connection's token in the path. Protocol version 2024-11-05. The server advertises tools and nothing else: no prompts, no resources, no sampling.

```
https://brassdeck.com/mcp/<your token>
```

Connecting

Sign in, open **Connectors**, choose Claude or ChatGPT, and decide before you press Connect whether it may edit. You are shown a URL once. Paste it into the assistant.

- **In Claude:** Settings, Connectors, Add custom connector, paste the URL.
- **In ChatGPT:** Settings, Connectors, Add, paste the URL.

One token per connection. Connect both and you get two, so disconnecting one does not sign the other out. Each row on the Connectors page shows a hint from the token, whether it can edit, how many calls it has made and when it was last used.

Read-only is not a refusal, it is a shorter list. A read-only connection is never even shown the tools that write, so the assistant does not try one and then report the refusal as though the app were broken. Of the 15 tools, 5 read and 10 write.

Every tool

Read out of the server itself, so this table cannot fall behind it. Names are what an assistant would ask for rather than what the database calls it. Inputs marked required are the ones a call fails without.

TOOL	WHAT IT DOES	INPUTS	WRITES
list_projects	Every card on the Brassdeck canvas: what stage it is at, what the next step is, how long since it last moved, and whether it is stalled. `steps_left` is what remains (up to six) and `last_done` is the most recent finish. Start here — one call gives the whole picture and avoids a round trip per card; use <code>get_project</code> for the full list.	stage STRING Optional filter: spark, shaping, building, live, parked or dropped. stalled_only BOOLEAN Only cards with nothing decided as a next step. include_finished BOOLEAN Include cards already marked done. Off by default, because finished work is not outstanding work. Turn it on to answer "what did I actually finish". include_notes BOOLEAN Include notes. Off by default. A note has no stage and no clock — it is something written down to stop holding it, not work in progress.	no
recent_progress	What actually moved, day by day, over the last few days: steps finished, next steps decided, stage changes, cards parked, dropped, revived or finished. Adding a step is planning and is not in here. This is the tool for "what did I finish this week", "what happened since Monday", "what moved today". Without it there is no way to answer those, so do not answer them from <code>list_projects</code> , which only carries the latest finish per card.	days INTEGER How many days back, 1 to 90. Default 7.	no

TOOL	WHAT IT DOES	INPUTS	WRITES
get_project	One card in full, including every step in the order it was written. Use this when asked to work THROUGH something — list_projects only carries the first six steps.	slug STRING, REQUIRED The card slug, from list_projects.	no
search	Find cards by any word in their name, summary, next step or any step written on them.	query STRING, REQUIRED Words to look for. All of them must appear.	no
create_project	Put a new card on the canvas. Use for an idea worth keeping, not for a task.	name STRING, REQUIRED What to call it. next_action STRING Optional. The single next thing to do. summary STRING Optional. One line on what it is.	yes
create_routine	Set up a standing notification: when something on the board reaches a threshold, email the account. Use this when someone says "tell me when", "let me know if", "notify me once". It fires ONCE and then stops. ALWAYS say back in plain words what you set up, including the exact number — a routine nobody knows was created is worse than none.	kind STRING, REQUIRED video_views for a YouTube link on a card, domain_free for a domain name on a card, project_quiet for a card nobody has touched. subject STRING, REQUIRED For video_views, the YouTube link or its 11 character id. For domain_free, the domain. For project_quiet, the card slug. threshold INTEGER Views for video_views. Whole days for project_quiet. Omitted for domain_free. note STRING What the person actually asked for, in their words. Quoted back in the email.	yes

TOOL	WHAT IT DOES	INPUTS	WRITES
list_routines	Every standing notification on this board, including ones that have already fired. Use it when someone asks what they are being told about.	none	no
delete_routine	Stop a standing notification. Takes the id from list_routines.	id INTEGER, REQUIRED From list_routines.	yes
write_note	Write something down as a NOTE, which is a thing to remember rather than work being carried. A note has no stage, no next step and no clock, so it never appears as something outstanding and never goes cold. Use this for anything the person wants kept but is not going to move along: a number, an address, what somebody said, a passage worth having later. Use create_project instead for an idea they actually intend to take somewhere.	name STRING, REQUIRED A short title, so it can be found on the canvas. body STRING The writing itself. Line breaks are kept.	yes
add_step	Write a step onto a card. Adding steps is planning, so it does NOT count as progress.	slug STRING, REQUIRED The card. title STRING, REQUIRED The step.	yes
complete_step	Tick a step off. This DOES count as progress and resets the card's clock.	slug STRING, REQUIRED The card. title STRING, REQUIRED The step title, or enough of it to match one.	yes
set_next_action	Set the one next thing for a card. A card with nothing here is what this app calls stalled, so this is often the single most useful change to make.	slug STRING, REQUIRED The card. next_action STRING The next thing. Empty clears it.	yes

TOOL	WHAT IT DOES	INPUTS	WRITES
set_star	Star or unstar a card. A star means "I am on this right now" — it is one bit, not a priority level, and there is no ranking. Starring is NOT progress: it does not touch how long since the card last moved, and it must never be described as if it were work done. Parking or dropping a card clears its star.	slug STRING, REQUIRED The card. starred BOOLEAN, REQUIRED true to star, false to unstar.	yes
set_waiting	Record that a card is blocked on somebody else, or that it no longer is. Pass who or what is being waited on — "Wise support", "the App Store review queue", a person. Pass an empty string to end the wait. WAITING IS NOT A STAGE and does not change one. A card waiting on a bank is still at Building and stays there when the bank answers. It is also NOT progress in either direction: starting a wait is not work, and the reply landing was somebody else doing it, so the clock keeps running through the whole wait on purpose — a wait open for forty days is worth mentioning. What it does change is `stalled`, which goes false: somebody HAS decided what happens next.	slug STRING, REQUIRED The card. waiting_on STRING, REQUIRED Who or what is holding it. Empty string ends the wait.	yes

TOOL	WHAT IT DOES	INPUTS	WRITES
set_stage	Move a card along: spark (a thought), shaping (working out what it is), building, live. parked means deliberately stopped, dropped means decided against — both are kept, never deleted.	slug STRING, REQUIRED The card. stage STRING, REQUIRED spark, shaping, building, live, parked or dropped. reason STRING For parked or dropped: why it stopped, IN THE PERSON'S OWN WORDS. If they gave no reason, leave this out. Never invent one; it is written onto their card as their decision. until STRING For parked only: the date it comes back, YYYY-MM-DD. Convert "next month" or "in two weeks" to a date before calling.	yes

A tool that refuses answers normally and says why, rather than ending the exchange with an error. "Nothing here by that name" is a fact the assistant should read and act on, not a fault.

What it is told

The server does not only hand over a list of tools. Every connection is given a short briefing on the first call, so an assistant that has never seen this app does not invent its own words for it. In summary, it is told:

- **The vocabulary, and to use it exactly.** Card, board, canvas, step, stage, and the two exits. It is told which words it may never say.
- **That this is not a list of things owed.** A card is an intention worth keeping, and stalled means only that nobody has decided a next step yet, which is a question rather than a failure.
- **That the clock is honest and must stay that way.** Which actions move it and which do not, and that the number is reported plainly: never as a reproach, never with a target, never as a streak.
- **That parked and dropped are kept forever.** Never to suggest deleting them, never to count them as outstanding, never to call stopping something on purpose falling behind.

- **That it is your board.** If you name a card it has not seen, search before doubting you, and call a tool rather than answering from memory.
- **Whether it may write.** A writing connection is also told it may **offer** once, in one sentence, to put something on your board when you are clearly starting something with a life beyond the conversation, and to create it only if you say yes. Never unasked. A read-only connection is simply told it is read-only.

Tokens and revoking

- **The token rides in the path.** That is the one field every assistant has: you paste a URL. It is 32 bytes from a cryptographic random source, held to exactly the rules a share link is held to.
- **Only its hash is stored.** Nobody, including whoever runs this app, can print a live connection URL back to you. That is why it is shown once.
- **Revoke is a column, not a delete.** Disconnecting marks the row and keeps it, with the call count, because what once had access to your board is part of the record.
- **Every tool is scoped from the token.** The workspace and the member come off the token's own row. Nothing an assistant sends can name a workspace, which is the difference between an assistant that can read your board and one that can read everybody's.
- **A connection is a key.** Anyone holding the URL has the access it was made with. Treat it the way you treat a share link, and disconnect anything you are not using.

If this page and the app disagree, the app is the bug.

Say so, and it gets fixed.
