



CORSO IA CON CLAUDE · DISPENSA PER L'ALLIEVO

Artifacts e generazione di applicazioni web

Quando la chat smette di essere una conversazione e diventa un banco di lavoro

ORA	DURATA	MODULO
07 di 12	60 minuti	4 — Strumenti

Gianluca Bonomo · www.gianlu.it · info@gianlu.it



Indice

Indice 2
Come usare questa dispensa 3
1. Che cos'è un Artifact 4
1.1 Che cosa diventa un Artifact 5
1.2 Che cosa non è 5
2. A che cosa serve a chi non programma 6
2.1 Il prototipo da far vedere in riunione 6
2.2 Il calcolatore da usare o da mandare 6
2.3 Lo schema che si corregge in diretta 6
2.4 Il documento che si riscrive davanti a te 6
2.5 Quando conviene un Artifact e quando no 7
3. Come si chiede bene 7
3.1 Il ciclo di correzione, su un caso vero 8
3.2 Le frasi che sbloccano una correzione difficile 9
4. Che cosa si può mettere in produzione 9
4.1 Condividere il risultato 9
4.2 I limiti tecnici da conoscere 10
4.3 Che cosa consegnare a uno sviluppatore 10
5. Esercitazione pratica dell'ora 11
5.1 Che cosa provare per primo 11
Le domande che arrivano in aula 12
Glossario 13
Per approfondire 14

Come usare questa dispensa

Questa dispensa raccoglie, in forma scritta e ordinata, tutto quello che vediamo insieme durante l'ora di lezione. Non devi studiarla a memoria: serve a farti ritrovare le cose con calma, dopo, quando l'aula è finita e ti resta una domanda in sospeso.

È pensata per chi parte da zero. Ogni termine tecnico viene spiegato la prima volta che compare, e a fine dispensa trovi un glossario con tutte le parole nuove. Se una pagina ti sembra difficile, saltala e torna indietro dopo: nessun concetto di questa dispensa richiede conoscenze informatiche precedenti.

> **COME SONO FATTI I RIQUADRI**
>
> **In parole semplici** (azzurro): la spiegazione senza tecnicismi, quella da portarsi a casa se si ricorda una cosa sola.
>
> **Attenzione** (oro): un errore frequente, un rischio, una cosa da non fare.
>
> **Esempio** (grigio): un caso concreto, preso dalla vita di tutti i giorni o dal lavoro in azienda.

Obiettivo dell'ora

Imparare a usare gli Artifacts per generare e vedere in tempo reale pagine web, grafica, diagrammi e documenti, e capire a che cosa servono davvero in azienda a chi non programma.

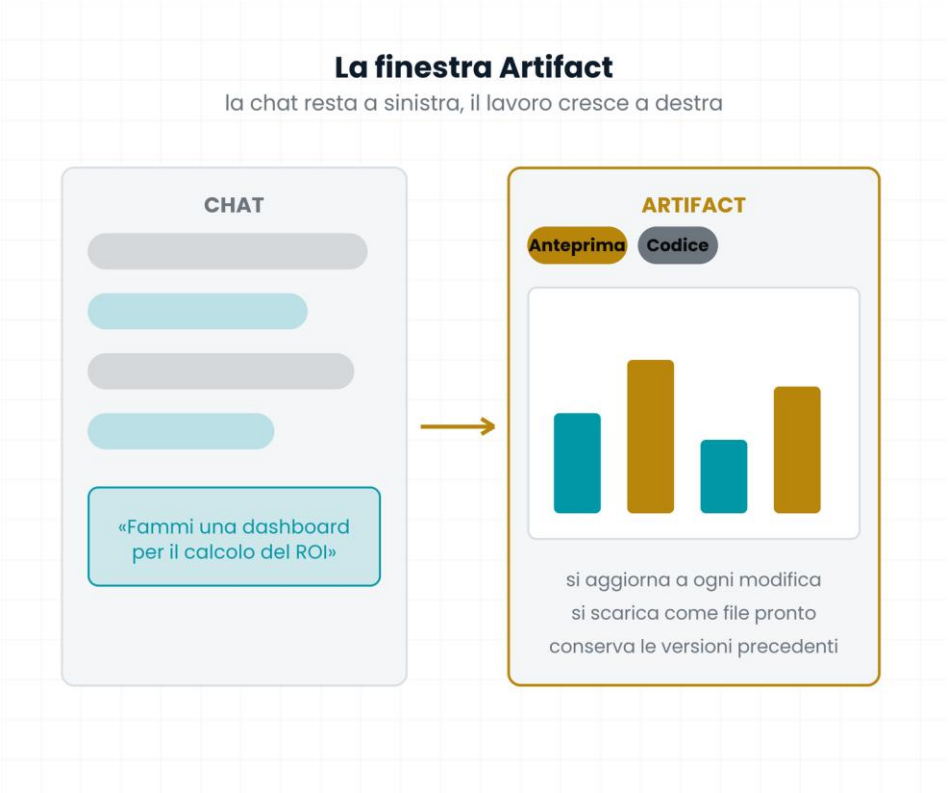
Che cosa saprai fare alla fine

- Riconoscere quando si apre un Artifact e perché.
- Chiedere un Artifact in modo che esca utile al primo colpo.
- Correggerlo per passi successivi senza ricominciare da capo.
- Capire che cosa si può mettere in mano a un collega e che cosa no.

1. Che cos'è un Artifact

È il momento in cui il risultato smette di scorrere via nella conversazione e diventa un oggetto su cui si lavora.

Quando chiedi a Claude qualcosa di complesso — una pagina web, del codice, un grafico, un documento lungo — il risultato non finisce nel flusso della chat. Si apre un **pannello dedicato a fianco**: l'Artifact.



A sinistra si continua a parlare, a destra il lavoro cresce.

Dentro quel pannello puoi passare fra **anteprima** e **codice** con un clic, vedere il risultato aggiornarsi a ogni modifica che chiedi, tornare alle versioni precedenti e infine scaricare un file pronto da usare.

PERCHÉ CAMBIA IL MODO DI LAVORARE

Senza Artifact, il risultato è testo: lo leggi, lo copi, lo incolli altrove, e se non va bene ricominci.

Con l'Artifact il risultato è una **cosa**: la guardi, la provi, chiedi una modifica e la vedi cambiare. È più vicino al lavorare che al chiedere.

1.1 Che cosa diventa un Artifact



Sei tipi di contenuto che aprono una finestra dedicata.

In sintesi: pagine web funzionanti, codice sorgente di una certa lunghezza, grafica vettoriale, diagrammi di flusso, documenti lunghi, tabelle e strutture dati.

1.2 Che cosa non è

Vale la pena togliere subito tre equivoci, perché sono quelli che portano alle delusioni più grosse.

Non è...	Perché
un'applicazione aziendale	Non ha un database, non gestisce utenti, non conserva i dati fra una sessione e l'altra se non glielo si chiede esplicitamente
un sostituto dello sviluppatore	Fa benissimo la prima versione; la manutenzione, la sicurezza e l'integrazione con i sistemi aziendali restano un mestiere
un documento definitivo	È pensato per essere corretto in diretta. Se lo trattate come un file da consegnare, ne sprecate la parte migliore

Detto questo, la maggior parte del lavoro d'ufficio non ha bisogno di un'applicazione aziendale: ha bisogno di uno strumento che funzioni stamattina, per tre persone, su un problema preciso. È esattamente lo spazio in cui un Artifact è imbattibile.

2. A che cosa serve a chi non programma

Questa non è una funzione per sviluppatori. È una funzione per chi deve far capire un'idea in fretta.

2.1 Il prototipo da far vedere in riunione

Descrivere a parole come dovrebbe funzionare uno strumento richiede dieci minuti e lascia metà sala con un'idea diversa. Farlo vedere richiede lo stesso tempo e non lascia dubbi.

2.2 Il calcolatore da usare o da mandare

Un simulatore di preventivo, un calcolo di ritorno dell'investimento, una tabella interattiva: cose che oggi si fanno con un foglio di calcolo complicato, e che diventano una pagina che chiunque sa usare.

2.3 Lo schema che si corregge in diretta

Un diagramma di processo o un organigramma generato dalla descrizione a parole, e modificato mentre lo si guarda insieme. È il modo più rapido per mettere d'accordo delle persone su come funziona davvero una procedura.

2.4 Il documento che si riscrive davanti a te

L'uso meno appariscente è anche quello che si rivela più utile ogni giorno: un testo lungo — un'offerta, una procedura, una relazione — che resta aperto in un pannello a fianco e si modifica a comando, senza essere riscritto da capo ogni volta.

La differenza rispetto alla chat normale è concreta: chiedendo «accorcia il terzo paragrafo» in una conversazione ordinaria, la risposta ripete tutto il documento e voi dovete ricopiarlo. Con un Artifact cambia solo quel paragrafo, e il resto resta com'era, parola per parola.

- «Rendi il tono più diretto, ma non toccare la parte sui prezzi.»
- «Aggiungi una premessa di cinque righe per chi non conosce il progetto.»
- «Riscrivi le conclusioni pensando a un lettore che deve decidere, non capire.»
- «Trasformalo in un elenco puntato, tenendo le stesse informazioni.»

> **PERCHÉ CONTA PIÙ DI QUANTO SEMBRI**
>
> Perché è il modo in cui si scrive davvero: per revisioni successive, non di getto.

E perché ogni revisione resta una versione: potete confrontare, tornare indietro, riprendere una formulazione scartata due giri prima.

2.5 Quando conviene un Artifact e quando no

Non tutto merita di diventare un Artifact. La discriminante è se il risultato **serve come oggetto** o serve solo come testo da leggere.

Conviene	Non conviene
Il risultato va mostrato a qualcuno, o usato più volte	È una risposta che leggete e basta
Prevedete di correggerlo a più riprese	Lo volete così com'è, subito
Deve avere una forma precisa: una pagina, uno schema, un calcolo che risponde	È un testo discorsivo breve
Volete poterlo scaricare e passare a un collega	Resta dentro la conversazione

Nel dubbio, si può sempre chiedere di trasformare una risposta già ottenuta: «fammene una pagina che posso mostrare in riunione». Non si riparte da capo.

3. Come si chiede bene

Invece di…	Scrivi…	Perché
«Fammi una dashboard»	«Fammi una pagina che calcoli il ritorno dell'investimento a partire da costo dello strumento, ore risparmiate e costo orario»	Il modello deve sapere che cosa deve **fare**, non che aspetto deve avere.
«Falla bella»	«La userà un commerciale al telefono: pochi campi, caratteri grandi»	Dire chi la userà vale più di qualsiasi indicazione estetica.
Venti richieste in un colpo	Una per volta: «ora aggiungi un grafico», «ora rendila stampabile»	Ogni modifica genera una versione: si può sempre tornare indietro.

IL CICLO GIUSTO

Chiedi la cosa minima che funziona. Guardala. Chiedi **una** modifica. Guardala di nuovo.

In quattro giri arrivi molto più lontano che con un prompt lunghissimo al primo tentativo.

Prima di vedere il ciclo completo, conviene fissare le quattro cose che, dette nella prima richiesta, fanno risparmiare due giri di correzioni.

1. **Chi lo userà.** «Lo userà un commerciale al telefono» cambia tutto: caratteri grandi, pochi campi, nessun termine tecnico.
2. **Che cosa entra e che cosa esce.** I dati da inserire e il risultato atteso, elencati. È la parte che le persone danno per scontata ed è quella che serve di più.
3. **Su che cosa gira.** «Deve funzionare sul telefono» o «va proiettato in riunione» portano a due risultati molto diversi.
4. **Che cosa non deve fare.** «Non deve chiedere dati personali», «non deve salvare niente»: i vincoli detti prima valgono dieci correzioni dopo.

3.1 Il ciclo di correzione, su un caso vero

Il modo più efficace di lavorare con un Artifact non è scrivere il prompt perfetto: è arrivarci in quattro giri. Ecco una sequenza reale, su un calcolatore del ritorno dell'investimento.

Giro	Che cosa si chiede	Che cosa si ottiene
1	«Fammi una pagina che calcoli in quanti mesi rientro da un investimento, partendo da costo mensile, ore risparmiate a settimana e costo orario»	Una pagina funzionante ma spoglia: tre campi e un numero.
2	«Aggiungi un grafico che mostri il risparmio cumulato nei primi dodici mesi»	Lo stesso calcolatore, con il grafico. La versione precedente resta raggiungibile.
3	«Usa i colori aziendali: intestazioni in blu petrolio, accenti in oro. Caratteri grandi, la userà un commerciale al telefono»	L'aspetto si allinea al marchio, e la leggibilità migliora.
4	«Aggiungi un pulsante per stampare e un campo per il numero di persone coinvolte»	Lo strumento diventa utilizzabile davvero, e si può scaricare.

PERCHÉ FUNZIONA MEGLIO DI UN PROMPT LUNGHISSIMO

Perché a ogni giro **vedete** il risultato e correggete la rotta. Con un prompt unico da venti righe, se qualcosa non torna non sapete quale delle venti richieste sia stata fraintesa.

E poiché ogni modifica genera una versione, tornare indietro di un passo costa un clic.

3.2 Le frasi che sbloccano una correzione difficile

Capita di girare a vuoto: chiedete una modifica, arriva qualcosa che non è quello che volevate, riprovate e peggiora. Quattro formulazioni risolvono quasi sempre l'impasse.

Quando succede che…	Provate a dire
a ogni correzione si rompe qualcos'altro	«Cambia **solo** questo, lascia tutto il resto esattamente com'è»
non capite più che cosa ha cambiato	«Prima di modificare, dimmi in tre righe che cosa hai intenzione di cambiare»
il risultato è diventato troppo complicato	«Torna alla versione di due passaggi fa e riparti da lì»
non riuscite a spiegare che cosa non va	«Ecco che cosa vedo, ecco che cosa mi aspettavo di vedere» — e descrivete le due cose separatamente

IL PRINCIPIO DIETRO TUTTE E QUATTRO

Una correzione funziona quando è **piccola e circoscritta**. Chiedere tre cose insieme, a metà del lavoro, è il modo più rapido per perdere il controllo del risultato.

4. Che cosa si può mettere in produzione

IL LIMITE DA CONOSCERE

Per un prototipo, uno strumento interno o una simulazione, quello che esce è già utilizzabile.

Per qualcosa che tratta dati di clienti, gestisce pagamenti o va pubblicato verso l'esterno, serve una **revisione tecnica** vera. Un Artifact non sostituisce il lavoro di uno sviluppatore: gli fa risparmiare la parte noiosa.

C'è però un valore che vale la pena mettere a fuoco: arrivare da un fornitore con un prototipo che mostra il comportamento voluto, invece che con un documento di requisiti, accorcia i tempi e riduce i malintesi in modo sorprendente.

4.1 Condividere il risultato

Un Artifact si scarica come file e da quel momento vive per conto suo: si apre in un browser qualsiasi, senza Claude e senza collegamento al servizio. È il modo più semplice per passarlo a un collega o allegarlo a un messaggio di posta.

PRIMA DI MANDARLO FUORI DALL'AZIENDA

Controllate che dentro non siano rimasti i dati veri usati per le prove: nomi di clienti, importi, listini. Capita più spesso di quanto si creda.

E ricordate l'obbligo di trasparenza visto nella seconda ora: se un contenuto generato dall'IA viene diffuso all'esterno, la sua natura va dichiarata.

4.2 I limiti tecnici da conoscere

Tre limiti tornano sempre, ed è meglio conoscerli prima di promettere qualcosa a un collega.

- **Non parla con i vostri sistemi.** Un Artifact non si collega al gestionale né all'archivio aziendale: i dati glieli date voi, scrivendoli o incollandoli. Per il collegamento vero servono le integrazioni dell'undicesima ora.
- **Non conserva nulla da solo.** Chiudendo la pagina, quello che una persona ha inserito mentre lo usava non resta, a meno che non sia stato previsto esplicitamente un salvataggio nel browser.
- **Cresce fino a un certo punto.** Oltre una certa complessità conviene fermarsi: non perché non ci riesca, ma perché a quel punto state costruendo un software vero, e un software vero va progettato.

IL MOMENTO IN CUI FERMARSI

Quando cominciate a chiedere «fai in modo che più persone possano usarlo insieme» o «collegalo al nostro archivio», il prototipo ha finito il suo lavoro.

Ha fatto la cosa più difficile: ha chiarito che cosa serve. Da lì in poi serve qualcuno che lo costruisca per davvero.

4.3 Che cosa consegnare a uno sviluppatore

Quando il prototipo ha fatto il suo lavoro e si passa alla versione vera, il modo in cui lo si consegna fa la differenza fra due settimane e due mesi.

1. **Il file dell'Artifact**, scaricato. Vale più di qualsiasi descrizione: si apre e si vede come deve comportarsi.
2. **L'elenco di che cosa è vero e che cosa è finto.** Nel prototipo i dati sono inventati e i calcoli semplificati: va detto quali sono le regole reali.

3. **I casi limite che vi interessano.** Che cosa deve succedere se un campo è vuoto, se un valore è negativo, se il risultato non è calcolabile.
4. **Chi lo userà e quanti sono.** È l'informazione che determina metà delle scelte tecniche, e quasi nessuno la scrive.

> **IL VANTAGGIO VERO DEL PROTOTIPO**
>
> Non è il codice: quello verrà riscritto quasi sempre.
>
> È che la discussione su «che cosa deve fare» avviene guardando una cosa che si muove, invece che leggendo un documento che ognuno interpreta a modo suo.

5. Esercitazione pratica dell'ora

Una dashboard per il calcolo del ritorno dell'investimento, costruita da zero partendo da una frase.

1. Chiedete una pagina con tre campi in ingresso: costo mensile dello strumento, ore risparmiate a settimana, costo orario medio.
2. In uscita: risparmio mensile, mesi necessari per rientrare, un grafico.
3. Poi una modifica per volta: i colori aziendali, un pulsante per stampare, un campo per il numero di persone coinvolte.
4. Scaricate il file. Funziona anche senza Claude: è vostro.

> **UN'IDEA IN PIÙ, SE AVANZA TEMPO**
>
> Usate la stessa pagina nella dodicesima ora, come parte del progetto finale: è esattamente il tipo di risultato che si presenta bene.

5.1 Che cosa provare per primo

Se dovete scegliere una cosa sola da provare dopo il corso, sceglietene una con queste tre caratteristiche: la conoscete bene, serve a poche persone, e oggi si fa con un foglio di calcolo scomodo.

- Il calcolatore che il commerciale usa al telefono per dare un ordine di prezzo.
- La scheda che il tecnico compila a fine intervento, oggi su carta.
- Il modulo interno che qualcuno riempie a mano e poi ricopia da un'altra parte.

- Il grafico che ogni mese si rifà partendo dagli stessi dati.

Sono tutti casi in cui il risultato si vede subito, non ci sono dati delicati in gioco e il confronto con il modo attuale di lavorare è immediato. È il tipo di prova che vi fa capire in mezz'ora se lo strumento vi serve o no.

Le domande che arrivano in aula

Sono le stesse in ogni corso. Vale la pena avere pronta una risposta breve e onesta, anche per quando le farà un collega.

«Posso metterlo davvero in produzione?»

Per un prototipo, uno strumento interno o una simulazione, sì: quello che esce funziona.

Per qualcosa che tratta dati di clienti, gestisce pagamenti o va pubblicato all'esterno serve una revisione tecnica vera. Un Artifact non sostituisce uno sviluppatore: gli risparmia la parte noiosa.

«Non so programmare: posso comunque modificarlo?»

Sì, ed è anzi il modo previsto: la modifica si chiede **a parole**, non si tocca il codice.

Il pannello del codice serve a chi lo sa leggere; per tutti gli altri ci sono l'anteprima e la conversazione.

«Se sbaglio una modifica perdo il lavoro fatto?»

No: ogni modifica genera una nuova versione, e le precedenti restano raggiungibili. Si torna indietro di un passo e si riparte da lì.

«A che serve, se poi lo rifà un fornitore?»

A far capire esattamente che cosa volete. Presentarsi da un fornitore con un prototipo che si comporta come deve, invece che con un documento di requisiti, accorcia i tempi e riduce i malintesi in modo sorprendente.

È il passaggio da «vorrei qualcosa del genere» a «deve fare questo».

Glossario

Le parole che incontri in questa dispensa, spiegate in una riga. Tienilo a portata di mano: torneranno in tutte le ore successive del corso.

Termine	Che cosa significa
Artifact	Il pannello laterale in cui Claude costruisce contenuti complessi: pagine, codice, grafica, documenti.
Anteprima	La vista che mostra il risultato funzionante, invece del codice che lo genera.
HTML, CSS, JavaScript	I tre linguaggi con cui è fatta una pagina web: struttura, aspetto, comportamento.
Mermaid	Modo testuale di descrivere diagrammi di flusso e schemi, che vengono poi disegnati automaticamente.
Prototipo	Una versione funzionante ma non definitiva, fatta per capire se un'idea sta in piedi.
SVG	Formato di grafica vettoriale: si ingrandisce senza perdere qualità.

Per approfondire

Le fonti usate per costruire questa dispensa. Sono tutte pubbliche e consultabili: se un dato ti interessa, vai alla fonte e verificalo. È esattamente quello che chiediamo di fare anche all'intelligenza artificiale.

La documentazione sugli Artifacts

Il centro assistenza di Claude descrive quali contenuti aprono un Artifact, come si gestiscono le versioni e come si condividono.

Il capitolo sul prompting pratico

Niente paura, è solo IA, capitolo 6.6, raccoglie esempi di richieste per creatività, studio, lavoro e vita quotidiana.

Il libro di riferimento del corso

Gianluca Bonomo, *Niente paura, è solo IA. La guida completa per capirla, usarla e restare umani*.

UNA NOTA SULLE DATE

Le norme sull'intelligenza artificiale cambiano in fretta. I dati normativi di questa dispensa sono verificati a **settembre 2026**. Prima di usarli in una decisione aziendale, controlla sempre la fonte ufficiale: il calendario dell'AI Act è già stato modificato una volta.