

ISO 26517 readiness kit (specimen)

ISO/IEC/IEEE DIS 26517, Systems and software engineering: Development of user assistance in mobile applications. Edition 1, stage 40.60, close of voting on 25 June 2026.

Planning date	14 September 2027. An estimate from the stage, not an ISO announcement.
Record read	2026-10-21
Public basis	ISO record 87297 and the free ISO Online Browsing Platform preview.
Not public	The requirement wording of clauses 4 to 9 and Annex A. The prepare and evidence columns are preparation guidance.

This kit is a register you can run. It carries one row for every heading and defined term in the public contents, a preparation note, an evidence slot, an owner, a status and a due date. The PDF gives the board brief and the plan; the CSV and XLSX carry the working register.

This is a labelled specimen. It carries four register rows and a shortened brief. The paid kit carries all 80 rows, 46 evidence items, the full plan and the method note.

SPECIMEN

The register (sample)

Ref	Heading	What to prepare	Evidence
ISO26517-4.2	Defining the target audiences	First time users, expert users and the groups in between, named before design. Write a help plan with the app roadmap, the release dates, the owners and the help forms for each release.	The user assistance plan with owners, releases and help forms
ISO26517-6.3.4	Chatbots	Conversational help, with the boundary, the fallback and the record it keeps. Draw a journey map for each app and mark the moments where help is needed, from first launch to a repeated expert task.	A help journey map with the chosen form at each moment
ISO26517-7.1.2	Writing in steps	One action per step, in the order a person performs it. Write a mobile help style guide with a section for steps, colour, icons, simple language and gestures.	A mobile help style guide with the clause 7 sections
ISO26517-9.2	Text description of an image	A text alternative for every image that carries meaning. Inventory every image, screenshot, icon and video in the help and record whether it carries meaning.	A register of every image with meaning, description and owner

These are the four rows the free specimen page names: 4.2, 6.3.4, 7.1.2 and 9.2. The paid register carries all 80 rows with owner, status and due date columns.

SPECIMEN

The board brief

What the draft is

ISO/IEC/IEEE DIS 26517 is edition 1 of a new international standard for user assistance in mobile applications. It sets requirements and recommendations for designing, developing and evaluating help inside phone and tablet apps. It is not a revision of 26514 and it does not replace any published 2651x standard. Clause 2 names ISO/IEC/IEEE 26514:2022 as its single normative reference.

Where it stands

The DIS was registered on 23 January 2026. The ballot opened on 1 April 2026 and closed on 25 June 2026. The record stands at stage 40.60. The next public checkpoints are 40.99, approval for registration as an FDIS, then 50.00, 50.20, 50.60 and finally 60.60, publication. No publication date has been announced.

What the public contents shows

Planning and analysis	Clause 4: the plan, the target audiences, the business use case, and the mobile context of platform, screen, physical environment and security.
Design principles	Clause 5: minimalism, four C principles, progressive disclosure. The four elements of the C principles are not in the free preview.
Designing content	Clause 6: first use, embedded help, and other forms from a help app and video to a chatbot, email support, social channels and AI.
Developing content	Clause 7: small screen writing, gesture language, variables and conditional text, illustrations and video, translation.
Evaluating	Clause 8: testing goals, measures and mobile testing.
Accessibility	Clause 9: accessible help, with a specific subclause on the text description of an image.

The decision for the board

The work that survives a change of wording is the work to start now: the scope, the audiences, the mobile context, the help form map, the style guide, the device test plan and the image descriptions. The kit turns those into a register with an owner and a date. The reading is a planning device, not a compliance claim.

1 Scope

The draft sets requirements and recommendations for help content in mobile apps, from first planning through to testing, and it leaves the device hardware itself out of scope.

What the public record shows

The free preview carries the full text of clause 1. Clause 1 is short and it decides every later argument about scope. It puts the draft in the middle of the product process rather than at the end of it: the help is designed, developed and evaluated, not bolted on after release. It names two routes into mobile help. One is content written for the mobile setting. The other is content that already exists for print, desktop or the web and is transformed by responsive design. The draft keeps the interface hardware out of its own remit. That exclusion is precise: a team still has to make the help readable on the screen it has, but the draft does not specify the screen, the speaker or the sensor.

What to put in place before publication

1. Write a one page scope statement for your user assistance that names the apps, the platforms and the languages covered.
2. List the help forms in play, such as embedded help, coach marks, a help site, a chatbot and video, and mark which are in scope this year.
3. Record which existing content will be transformed by responsive design and which will be written fresh for mobile.
4. State the exclusions plainly, including hardware specifications and any content owned by another team.
5. Name the person who owns the scope statement and the date it was last checked against the app roadmap.

The test a reviewer runs

- Does the plan say which apps and which help forms the work covers?
- Is there a written decision on responsive transformation, or is every piece of help assumed to be new?
- Can the team name what it is not doing, including the hardware side?
- Would a new product manager understand the boundary from the plan alone?

2 Normative references

One document is normative for this draft: ISO/IEC/IEEE 26514:2022, the design and development standard for information for users. A dated reference means that edition, not a later one.

What the public record shows

The free preview carries the full text of clause 2 and the single reference. Clause 2 is one line of substance and one rule of reading. The mobile standard does not restate the whole design process. It points at 26514:2022 for the process, the information structure and the general requirements, and it uses this draft for the mobile setting. That makes 26514 a required companion, not background reading. It also fixes the edition. The wording in the preview is the standard dated reference formula: only the edition cited applies. A team that owns the 2018 edition of 26514, or the 2008 edition of the old 26514, does not hold the referenced text.

What to put in place before publication

1. Get lawful access to ISO/IEC/IEEE 26514:2022 and record the edition and the date it was obtained.
2. Map your existing information development process to 26514:2022 rather than assuming the process is equivalent.
3. Note the places where 26514:2022 leaves a choice and where this draft makes a mobile specific demand.
4. Keep the reference list in the help plan, with the edition year and a link to the ISO record.
5. Remove undated or older citations from procedures that claim to follow the family.

The test a reviewer runs

- Does the team hold the 2022 edition, not an earlier one?
- Is the 26514 process mapped, or is the standard only named in a slide?
- Does the help plan name the mobile additions that sit on top of 26514?
- Will a reviewer find a dated reference in the procedure, not just in a training deck?

3 Terms and definitions

Nine terms fix the vocabulary for the draft, from coach mark to user experience, and eight abbreviations cover AI, AR, CSS, FAQ, GUI, JSON, STE and UI.

What the public record shows

The free preview carries the full text of clause 3, including every definition and abbreviation. The vocabulary is small and deliberate. User assistance is wider than a help page: it includes principles, design techniques and solutions. A coach mark is a small pop-up laid over the interface, not a tooltip in a web page. An icon is a screen graphic that stands for a function. Minimalism is not brevity for its own sake; it is critical information plus the least extra information needed to be complete. Responsive design is about adaptation to the device layout. Context of use brings in users, tasks, equipment and the physical and social setting. The abbreviations show the technical range of the draft: AI, AR, CSS, JSON, STE and UI all appear in the mobile help conversation.

What to put in place before publication

1. Put the nine terms and eight abbreviations into the house style guide with the same meaning as the draft.
2. Write a short usage note for user assistance, user experience and user interface so the team stops using them as synonyms.
3. Decide how you will name gestures and controls in help content, and keep that list beside the glossary.
4. Check translated glossaries for a matching term, not a literal translation.
5. Review the glossary when the published edition appears, because defined terms can change between a draft and publication.

The test a reviewer runs

- Does the style guide use user assistance in the same sense as the draft?
- Is a coach mark distinguished from a tooltip, a banner and a modal?
- Do the translated terms match the approved glossary?
- Are minimalism and brevity used as if they meant the same thing?

4 User assistance planning and analysis

Clause 4 asks a team to plan the help, name the audiences, analyse the business use case and record the mobile context before any content is written.

What the public record shows

The public contents lists clause 4 and its subclauses 4.1 to 4.4.5. Their text is not in the free preview, so this page does not state it. The public contents makes clause 4 the planning gate. It starts with planning for mobile devices, then asks for the target audiences, the use case, and the context of use. The context has five parts: a general statement, the technical platform, the screen size, the physical environment and security. That list is the most useful public signal in the draft for a team that is starting work now. It says the mobile setting is not one variable. A help design that works on a large phone, held in two hands, on a fast network, in a quiet room, is not automatically right for a small phone, held in one hand, outdoors, with a locked device. The clause sits before design and development, so a team that starts with the screens has already skipped it.

What to put in place before publication

1. Write a help plan with the app roadmap, the release dates, the owners and the help forms for each release.
2. Build an audience table that separates first time users, occasional users and expert users, with the tasks each group brings.
3. Write the business use case for each app and state what it means for the help: what a user must achieve, and what failure costs.
4. Complete a context sheet for each supported platform: operating system versions, screen sizes, network conditions, physical settings and security constraints.
5. Record the decisions the context forced, such as larger text, offline help, one handed use or a locked screen.
6. Set a review date for the plan that falls before the next app release, not after it.

The test a reviewer runs

- Can the team name the first time task and the expert task for the app?
- Is the business use case written down, or is it only in the product manager's head?
- Does the context sheet cover more than one device size and one operating system?
- Has anyone tested the help in the physical setting where the app is really used?
- Does the plan record what the security constraints allow the help to show or store?

5 Design principles

Clause 5 fixes the design stance: minimalism, four named C principles whose elements the preview does not state, and progressive disclosure.

What the public record shows

The public contents lists clause 5 and subclauses 5.1 to 5.4. Their text is not in the free preview. The four elements of the C principles are not stated, and this page does not guess them. The public contents gives clause 5 four design ideas and one warning. Minimalism is already defined in clause 3 as critical information plus the least extra information needed to be complete. Progressive disclosure is the technique of showing the next useful layer and keeping the rest reachable, and Annex A gives six named ways to do it, in seven subclauses. The four C principles are named but not expanded in the free preview. That gap matters. A team that fills the four Cs from memory, with confident, concise, clear and consistent, may be right or may be inventing a requirement. The honest position is to record the principle as named, keep the team's working definition in a separate note, and check the published text before relying on it.

What to put in place before publication

1. Write a design principle note for the help that adopts minimalism and progressive disclosure in your own words.
2. Record the team's working definition of the four C principles, clearly marked as a working note until the text is public.
3. Audit three screens and mark what is shown at first view and what is one layer deeper.
4. Set a rule for what must never be hidden behind a disclosure control, such as a safety instruction or a legal notice.
5. Give the design review a checklist item for each principle, so the review can fail a screen.
6. Keep the principle note short enough that a designer will read it before the design, not after.

The test a reviewer runs

- Can a designer say what the first view of a screen must contain?
- Is anything essential hidden behind a disclosure control?
- Is the four C note labelled as a working definition rather than a quotation?
- Has a reviewer looked at the help on the smallest supported screen?

6 Designing content

Clause 6 chooses the form of help for the moment: a first use tour or overlay, embedded help such as coach marks and guided help, or another channel such as a help app, chatbot or AI.

What the public record shows

The public contents lists clause 6 and subclauses 6.1 to 6.3.7. Their text is not in the free preview, so this page does not state it. Clause 6 is the most concrete part of the public contents and the most useful for a product team. It separates help at first use from help embedded in the interface and help delivered through another form. First use has a general subclause, a features tour and overlays. Embedded help has coach marks, guided help and embedded help. Other forms run from a standalone help application through infographics, chatbots, email support and social channels to the role of artificial intelligence. The range is a warning against treating user assistance as one artifact. A coach mark, a chatbot and a video answer different questions and carry different risks. The clause asks a team to choose deliberately, and the choice then drives the writing, the testing and the evidence.

What to put in place before publication

1. Draw a journey map for each app and mark the moments where help is needed, from first launch to a repeated expert task.
2. Choose the help form for each moment and record why the other forms were rejected.
3. Write a first run script that a person can leave at any point, and test the exit as carefully as the tour.
4. Set limits for overlays and coach marks: how many, how often, how they are dismissed and whether they return.
5. For a chatbot or AI help, write the boundary, the fallback to a person, the disclosure and the review of generated answers.
6. For every channel outside the app, record how the answer gets back into the product backlog.

The test a reviewer runs

- Can a user leave the first run tour without losing the task?
- Is there a limit on coach marks, or does every release add another one?
- Does the chatbot say when it does not know, and can a person take over?
- Is AI generated help reviewed before it reaches a user?
- Does each channel have an owner and a feedback path?

7 Developing content

Clause 7 turns the design into content: writing for a small screen, a gesture vocabulary, single source customization, illustrations and video, and translation.

What the public record shows

The public contents lists clause 7 and subclauses 7.1 to 7.5. Their text is not in the free preview, so this page does not state it. Clause 7 is the production clause, and its structure is a checklist for the help style guide. Writing for a small screen has its own general statement, then steps, colour, icons and simple language. The draft then asks for a language of touch gestures, which is unusual enough to be worth a separate style guide section. Content customization covers device specific variants, custom variables, conditional text and CSS, which points to a single source rather than copied files. Illustrations and video have a general statement, GUI representation, movement and process video, and augmented reality. Translation closes the clause. Read as a whole, clause 7 says the help is engineered: structured, conditional, translatable and kept in step with the interface it describes.

What to put in place before publication

1. Write a mobile help style guide with a section for steps, colour, icons, simple language and gestures.
2. Adopt one action per step and put the action first, then the result, then any warning.
3. Name every gesture the same way across help, release notes and support scripts.
4. Move repeated content into variables and conditional blocks, and test both branches of every condition.
5. Set a screenshot standard that matches the current interface, including dark mode and large text.
6. Build a translation kit with length limits, screenshots, glossary and a review step for the translated build.

The test a reviewer runs

- Does a step contain one action, written in the order the person performs it?
- Is colour ever the only signal in the help?
- Can a translator work from the kit without asking what a screenshot shows?
- Do the screenshots match the interface in the current release?
- Does a conditional block have a test for the branch that hides it?

8 Evaluating UA design and content

Clause 8 asks for testing goals, measures and mobile testing that reflect the device and the setting, not a desktop usability session relabelled as mobile.

What the public record shows

The public contents lists clause 8 and subclauses 8.1 to 8.3. Their text is not in the free preview, so this page does not state it. The public contents gives evaluation three steps. First set the content usability testing goals. Then define the measures and requirements that say whether the help worked. Then test for mobile applications. The order matters. A team that starts with a test script and looks for measures afterwards will report what it happened to observe. The mobile step is also a warning about method. A phone in a lab, on a fast network, with a moderator sitting beside the participant, is not the setting the help will meet. The draft leaves the exact measures open, so the team has to choose and defend them. That is an opportunity: completion, time, errors, findability and confidence can all be defined in a way that fits the app.

What to put in place before publication

1. Write the testing goals before the first session and tie each one to a decision the team can take.
2. Choose measures that fit the help, such as task completion, time to answer, wrong turns, findability and confidence.
3. Set a baseline on the current release so a later result can be compared with something.
4. Test on real devices and real networks, including a small screen, a slow connection and an interrupted session.
5. Include first time and expert users, and record which group each finding came from.
6. Keep consent, recording and personal data handling in the test plan, and delete what you do not need.

The test a reviewer runs

- Were the measures chosen before the test, or after the results?
- Did the test run on a real phone and a real network?
- Were first time and expert users both represented?
- Is there a baseline from the previous release?
- Did every finding get an owner and a decision?

9 Accessible UA for a mobile application

Clause 9 makes accessibility a property of the help itself, with a specific duty to give a text description of an image.

What the public record shows

The public contents lists clause 9 and subclauses 9.1 and 9.2. Their text is not in the free preview, so this page does not state it. The public contents gives accessibility its own clause, which is a stronger signal than a sentence inside the design clause. It has a general statement and one specific subclause on the text description of an image. That specific duty is the most concrete accessibility requirement visible in the public record, and it lands on the illustrations and screenshots that clause 7.4 already brings into the help. The general subclause is likely to carry the wider expectation: the help must be perceivable, operable and understandable with the assistive technology people actually use. The draft does not name a test method in the free preview, and this site does not invent one. Our preparation uses WCAG 2.2 AA as a working benchmark, labelled as ours, not as a claim about the draft.

What to put in place before publication

1. Inventory every image, screenshot, icon and video in the help and record whether it carries meaning.
2. Write a text description for each informative image that says what the image shows and why it matters.
3. Mark decorative images as decorative, so a screen reader can skip them.
4. Test the help with a screen reader, large text and a keyboard or switch control, on a real device.
5. Caption and transcribe video, and check that the player controls are reachable.
6. Use WCAG 2.2 AA as the working benchmark, and label it as the team's benchmark rather than a requirement of the draft.

The test a reviewer runs

- Does every informative image have a text description that reads as a sentence?
- Is the description about the meaning, not the file name or the colours?
- Can a screen reader user reach the help, read it and dismiss it?
- Is the help still usable at the largest text size the platform allows?
- Does every video have captions and a transcript?

A Examples of progressive disclosure techniques

Annex A is informative and shows six named ways to reveal help in layers: content preview, accordion panes, sliders, a hamburger icon, a wizard and advanced settings. They sit in seven subclauses, of which A.1 is general.

What the public record shows

The public contents lists Annex A and its seven subclauses A.1 to A.7: A.1 is general and A.2 to A.7 name six techniques. Their text is not in the free preview, so this page does not state it. Annex A is the practical companion to the progressive disclosure principle in clause 5.4. It collects six techniques in seven subclauses, each named plainly: a content preview, accordion panes, sliders, the hamburger icon, a wizard and advanced settings. The annex is informative, so it does not create a separate obligation. Its value is in the choice it forces. A content preview helps a person decide whether to read on. An accordion keeps a long answer inside a short screen. A slider moves through a range or a sequence. The hamburger icon opens a longer menu and needs a label. A wizard carries a long or risky task in steps. Advanced settings keep the deeper options out of the first view. The annex does not say that every technique is right for every app. It gives a vocabulary for a design review.

What to put in place before publication

1. List the disclosure techniques already used in the app and the help, and name each one with the annex vocabulary.
2. For each technique, write the user question it answers and the case where it is the wrong choice.
3. Check that the hamburger icon has an accessible name and that the menu it opens is reachable without a gesture.
4. Check that accordion and slider controls work with a screen reader and a keyboard or switch.
5. Make a wizard resumable and show progress, so a person can leave and return.
6. Keep advanced settings out of the first view but easy to find, and say in the help where they are.

The test a reviewer runs

- Can the team name each disclosure technique it uses?
- Does the hamburger icon have a text name, not just three lines?
- Can a keyboard or switch user open an accordion and move a slider?
- Can a person leave a wizard and return without losing the work?
- Is an advanced setting findable from the help, not only from a support call?

The register (sections)

Ref	Part	State	Rows
1	Scope	Public text	1
2	Normative references	Public text	1
3	Terms and definitions	Public text	12
4	User assistance planning and analysis	Headings only	10
5	Design principles	Headings only	5
6	Designing content	Headings only	17
7	Developing content	Headings only	19
8	Evaluating UA design and content	Headings only	4
9	Accessible UA for a mobile application	Headings only	3
A	Examples of progressive disclosure techniques	Headings only	8

The full CSV and XLSX carry 80 heading rows, 46 evidence items, the plan and the stage record.

The plan

When	Step	Outcome
Now	Name the owner and write the scope. One owner, one page, the apps and the help forms in scope, and the exclusions.	A signed scope statement and a named owner.
Within 30 days	Build the audience and context sheets. First time, occasional and expert tasks, and the platform, screen, physical and security context for each app.	An audience table and a context sheet per app.
Within 45 days	Choose the help form for each moment. A journey map with the chosen form at each moment, including the first run and the repeated expert task.	A help form map and a first run script with an exit.
Within 60 days	Write the mobile style guide. Steps, colour, icons, simple language, gestures, variables, conditional text, screenshots and video.	A style guide with a sample procedure and a screenshot standard.
Within 75 days	Set the measures and run one device test. Goals, measures, a baseline, real devices, first time and expert users, and a findings log with owners.	A test plan, a measures table and a findings log.
Within 90 days	Complete the image text descriptions. Every informative image, screenshot, icon and video with a description that says what it shows and why.	An image register and a screen reader test record.
Within 120 days	Prepare the translation kit. Glossary, length limits, screenshots and a translated build review.	A translation kit and a review record.
By 14 September 2027	Read the published text and re-date the plan. The planning date is an estimate. Replace every assumption with the published text, and confirm the delta.	A published-text map and a re-dated register.

The evidence list

The artefacts a reviewer or an internal assurance function will ask for, grouped by the part of the draft they belong to. The register carries an owner, a status and a pointer for each row.

Part	Category	Artefact
1	Scope	A signed scope statement naming apps, platforms, help forms and languages
1	Route	A list that separates content written for mobile from content transformed by responsive design
1	Exclusions	A note that names the device hardware and any out of scope content
1	Ownership	The named owner and the review date in the help plan
2	Held text	A licensed copy of ISO/IEC/IEEE 26514:2022 with its edition and access date
2	Process map	A clause map from 26514:2022 to your information development procedure
2	Reference list	The normative reference list in the help plan
2	Currency	A check that no procedure cites a superseded 26514 edition
3	Glossary	A glossary entry for each defined term, with the draft as its source
3	Usage	A usage note that separates UA, UX and UI in the style guide
3	Translation	The approved term list shared with translators
3	Currency	The glossary review date and the edition it was checked against
4	Plan	The user assistance plan with owners, releases and help forms
4	Audiences	An audience table with first time, occasional and expert tasks
4	Use case	A business use case note for each app in scope
4	Context	A context sheet covering platform, screen, environment and security
4	Decisions	A decision log that links each context finding to a design choice
5	Principles	A one page design principle note with the team's definitions
5	Disclosure	A layer map for each key screen: first view, second view, full text
5	Review	A design review record with a pass or fail for each principle
5	Working note	A dated note that marks the four C definitions as unverified
6	Journey	A help journey map with the chosen form at each moment
6	First use	A first run script with a dismiss and a skip path
6	Embedded help	Specifications for coach marks, guided help and embedded help
6	Other channels	A chatbot or AI boundary and escalation note
6	Feedback	A route from support messages back to the product backlog
7	Style guide	A mobile help style guide with the clause 7 sections
7	Steps	A sample procedure with one action per step and a visible result
7	Gestures	A gesture glossary used in the help and the support scripts
7	Single source	A content map that shows variables, conditions and device variants
7	Media	A screenshot and video standard, with current examples
7	Translation	A translation kit and a translated build review record
8	Goals	A test plan with goals written before the sessions
8	Measures	A measures table with definitions and a baseline
8	Devices	A device and network matrix used for the sessions
8	Findings	A findings log with severity, audience and a decision
8	Privacy	Consent and data handling records for the sessions
9	Image register	A register of every image with meaning, description and owner
9	Alt text	Approved text descriptions, reviewed with the image
9	Testing	Screen reader, large text and switch access test records on real devices
9	Media	Captions and transcripts for every video in the help

Part	Category	Artefact
9	Benchmark	A note that names WCAG 2.2 AA as our benchmark, not a draft requirement
A	Technique map	A map of the disclosure techniques used, named with the annex vocabulary
A	Choice	A note for each technique on when it helps and when it hinders
A	Accessibility	A test record for the hamburger, accordion, slider and wizard with assistive technology
A	Wizard	A resumable wizard with progress and a safe exit

SPECIMEN

The method note

Read first

The ISO record 87297 gives the designation, edition, page count, committee, ICS, price, stage and every dated stage transition. The free ISO Online Browsing Platform preview gives the complete contents, the foreword, the introduction, the scope, the single normative reference, nine defined terms, eight abbreviations and the bibliography. The IEEE project page gives the PAR date and the working group. The published family standards give context, and the ISO stage code page explains the codes.

Paraphrase, never reproduce

Every requirement and evidence line is written in our own words. Only titles, defined terms and data element names are kept exact, because they are the shared vocabulary. The record is checked with a seven word overlap test against the public source before it ships.

Say where the text stops

The free preview carries clause 1 to clause 3 in full. It lists clause 4 to clause 9 and Annex A but does not show their wording. The pages for those clauses say so, then carry preparation, evidence and a reviewer test. This kit does the same: the prepare and evidence columns are preparation guidance, not the wording of the draft.

Dates are labelled

The stage dates are public and exact. The planning date is an estimate from the stage, 328 days from the build date and about fifteen months after the close of the DIS ballot. If ISO publishes a date, the record is rebuilt and the estimate is replaced.

The limit

This is a reading, not an official interpretation. It does not certify, does not replace the text and does not give legal advice. Where a judgement is ours, it is labelled as ours.

Source: <https://www.iso.org/standard/87297.html> | Read: 2026-10-21