



Fieldra

Test Script

Step-by-step checks of every feature of the phone app and Fieldra Desktop,
with a practice site, its data, and the answers to expect.

Version 1.2 · October 2026 · Luminex Ltd

Use with the test data pack, Fieldra_Test_Data.zip, and the Fieldra Field Guide.



Contents

1. How to use this script	3
2. Phone: start, projects and the basics	6
3. Phone: Terrestrial Survey	8
4. Phone: field modules	11
5. Phone: Field Tools	15
6. Phone: map, layouts and export	18
7. Desktop: install, screen and projects	20
8. Phone and desktop together	21
9. Desktop: adding and exporting data	27
10. Desktop: editing and symbology	33
11. Desktop: select, filter, calculate, join	36
12. Desktop: overlay and feature tools	37
13. Desktop: network analysis	39
14. Desktop: toolbox and scripts	41
15. Desktop: rasters, TIN and hydrology	42
16. Desktop: statistics, sample design and chainage	48
17. Desktop: layouts, dashboard and photos	50
18. Drone mapping	54
19. Sign-off sheet	59

1. How to use this script

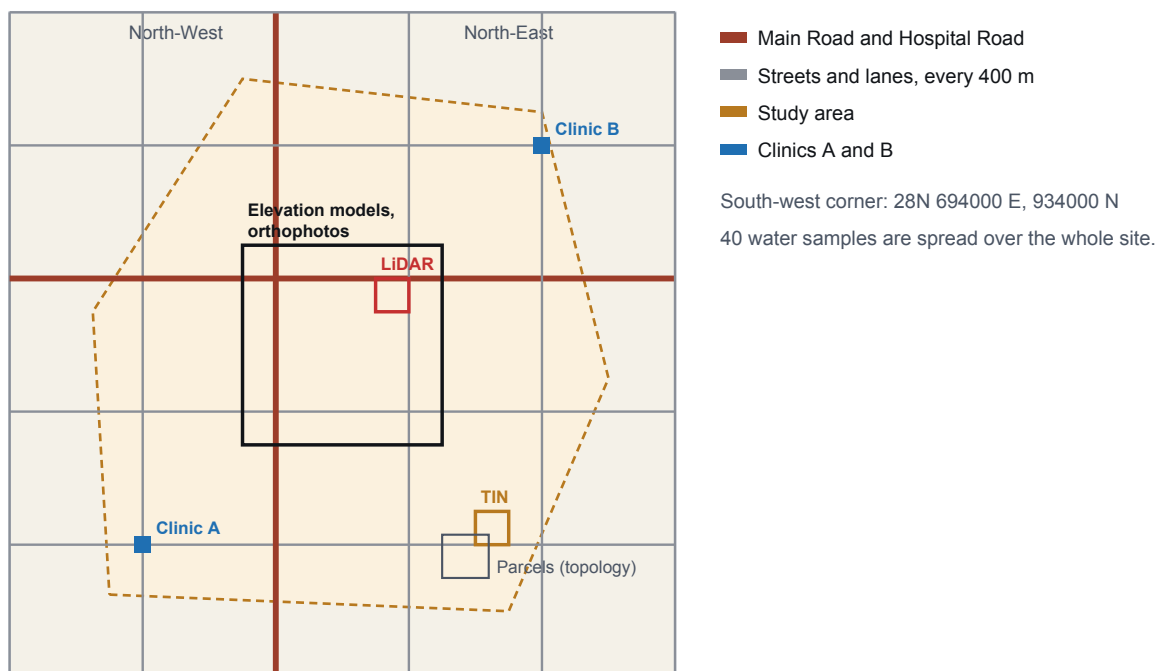
Work through the tests in order: later tests use layers made in earlier ones. Each test says what data it uses, the steps, and what you should see. Tick Pass or Fail and write anything odd under Notes; a sign-off sheet with every test is at the end.

What you need

Item	For
An Android phone or iPhone with Fieldra	Part A and B. A second phone is useful for phone-to-phone sending.
A Mac or Windows PC with Fieldra Desktop	Parts B to D. 16 GB of memory for photogrammetry.
Both on the same Wi-Fi (or a phone hotspot)	Sending between devices.
The test data pack, unzipped on the computer	Desktop tests. Copy a few files to the phone too (D-RTK points, planned points).
A 30 m tape (or any tape) and 4 pegs or stones	Land survey and area tests, outdoors.
Optional: an RTK receiver, Bluetooth water meter, drone photos, PostgreSQL	Tests marked Needs ; skip them if you don't have the equipment and write "not tested".

The practice site

The desktop data is a made-up site in **Freetown**, 2 km × 2 km, in WGS 84 / UTM zone 28N. Every file was made so that the right answers are known: a stockpile of exactly known volume, a plot with a known area, checkpoints with known errors, a LiDAR scene with two buildings and a power line, and so on. Where an answer can vary a little (rounding, cell sizes), the range is given.



The practice site: roads, districts, study area and where each data set lies.

The files

File	What it is
water_samples.csv	40 water sampling points: site_id, date (10 days, 4 a day), pH, turbidity, conductivity, condition, distance to the main road, rainfall, surveyor.
lab_results.csv, site_visits.csv	Tables without coordinates: lab results for 38 sites (W07 and W23 missing); visits to 10 sites (W01 three times, W02 twice).
districts.geojson, study_area.geojson	Four districts with population; a study area.
roads.geojson	60 road lines with name, class, speed_kmh (main 50, local 30), speed_am (main 15 in the morning rush) and oneway (one block of Main Road is TF: westbound only).
uturn_stops.geojson	Three route stops along one street, the last back towards the first.
clinics.geojson, households.geojson, candidate_sites.geojson	2 clinics, 30 households (people), 6 candidate sites.
orders.geojson, depot.geojson	12 delivery orders (load, service_min, not_before, arrive_by) and a depot with 2 vehicles of capacity 10.
spot_heights.csv, kerb_breakline.geojson	121 spot heights on a sloping plane (field z) and a kerb line (ELEV 23).
dsm_2026-01.tif, dsm_2026-03.tif	Elevation models 600 × 600 m at 1 m, two dates: two valleys, a stockpile that is taken away and another one tipped.
ortho_2026-01.tif, ortho_2026-03.tif	Orthophotos of the same area and dates.
multispectral_BGRN.tif, fields.geojson	A 4-band image (blue, green, red, near infrared) and the five fields in it.
stockpile_outline.geojson, section_line.geojson, pour_points.geojson	For volume, cross-section and watershed tests.
platform_dsm.tif	A graded platform, 120 × 80 m at 0.5 m, with three puddles.
site_lidar.las, site_lidar_no_crs.las	A LiDAR scene, 123,440 points: two buildings, eight trees, bushes, a power line and 40 low noise points; the second copy has no coordinate system.
checkpoints_surveyed.csv, checkpoints_model.csv	12 checkpoints as surveyed and as measured on the model.
gcps_drkt3.csv, planned_points.csv	For the phone: five GCPs as exported from DJI's app, and three planned points.
deed_traverse.txt	A plot boundary from a deed: start point, five bearings and distances.
parcels_topology.geojson	Seven parcels with one overlap, one gap, one self-crossing and one duplicate.
site_plan.dxf + .prj, boreholes.kml, walk_track.gpx	A CAD plan (4 layers), 3 boreholes, a walking track.
site_grid_points.csv	Four stations known in a site grid and in UTM.
household_water_xlsform.xlsx	A KoboToolbox / ODK form (XLSForm) with skip logic.

File	What it is
scanned_plan.png	A scanned plan with four marked crosses, to georeference.
script_example.py, script_example.js	Scripts for the scripting tests.
sl1968_shift.gsb	A datum shift grid (NTv2) for Sierra Leone 1968 (EPSG:4175) around Freetown: every point moves 1" in latitude and 2" in longitude, so the answers are easy to check.

Note: Phone tests in Part A are done where you are, not in Freetown: they use your own position, a tape and things around you.

2. Phone: start, projects and the basics

Install Fieldra on the phone first (store or the website's /beta page).

PH-01 Welcome survey and first project

What this tests: The very first use of the app: the short welcome questions and making the first project. A project is a folder that holds everything from one job or site.

What you get: A project called Test Site that every later phone test saves into.

1. Open Fieldra for the first time. Fill in the welcome survey (name, email, what you will use it for).
2. Create a project called **Test Site**. Allow location when asked.
3. Open Settings and type your name under Surveyor.

Expect: The welcome survey shows once only; the side menu shows Test Site at the top; Settings › Welcome survey says whether your answers were sent.

Pass ☐ Fail ☐ Notes _____

PH-02 Projects: switch, rename, delete

What this tests: That projects can be made, renamed and removed safely, and that removing one never touches another.

What you get: Test Site unchanged; the throwaway project and its record gone.

1. Side menu › Project › New: create **Throwaway**. Save any record in it.
2. Rename it to **Throwaway 2**. Try to make another project called Test Site.
3. Delete Throwaway 2 (press and hold it in the list).

Expect: The duplicate name is refused; deleting asks first and removes the project and its record; Test Site is untouched.

Pass ☐ Fail ☐ Notes _____

PH-03 Separate mobile sample project

What this tests: A saved fictional site for trying the normal record and map tools while keeping field projects separate.

What you get: An editable sample project that reopens without losing edits or adding duplicate records.

1. From Projects (or the first-run welcome page), tap Open sample site. Open Map and tap STREAM-01; edit its site ID to PRACTICE-01 and save.
2. Switch back to Test Site: open Map and Contents. Switch to Projects and Open sample site again.
3. Rename the sample project to Practice; reopen it through Open sample site. Optionally delete Practice after confirming it contains no work you need, then Open sample site again.

Expect: • The sample opens in Freetown and its points, tracks and areas open editable record details. Test Site has no sample features and no Example data toggle. • Reopening keeps PRACTICE-01 and the renamed project without duplicating it. Deleting then reopening creates a fresh sample with STREAM-01. Return to Test Site before the remaining field tests.

Pass ☐ Fail ☐ Notes _____

PH-04**Tablet sheets and larger text**

What this tests: Sheet width, wrapping labels and device text-size support on phones and tablets.

What you get: Readable labels and reachable controls in portrait, landscape and split view.

Needs: A tablet; repeat key text checks on a phone.

1. On an iPad or Android tablet, open map Contents and a long options sheet; rotate to landscape and try split view where available.
2. Increase the system text size. Reopen forms and options; scroll the map toolbar through every group. Restore the original system text setting afterwards.
3. On an iPhone 11 Pro Max or a similar phone, open an empty project: Environmental Survey › Observations, Water Survey and Records. Compare their empty messages with button labels and record content.

Expect: Sheets fill the app width and stay within safe areas. Long options and toolbar labels wrap; buttons remain reachable with larger text. Empty messages use the smaller supporting style consistently across modules.

Pass ☐ Fail ☐ Notes _____

PH-05**Position and accuracy colours**

What this tests: How the app shows how good the GPS position is. Indoors the phone can only guess its position to tens of metres; outside it is usually within a few metres. The colour tells the user whether to trust it.

What you get: You see the accuracy line change colour from indoors to outdoors.

1. Open any module's form indoors, then outdoors under open sky. Watch the accuracy line for a minute in each place.

Expect: Indoors: amber or red (worse than 5 m); outdoors: black (5 m or better) after a short wait.

Pass ☐ Fail ☐ Notes _____

PH-06**Saving and photos**

What this tests: The basic save: a record (one observation) with photos attached, and a copy of each photo in the phone's own gallery.

What you get: One saved record with one photo, and a Fieldra album in the phone's gallery.

1. In Environmental Survey › General, take a photo, add one from the library, remove one with ×, and save.
2. Open the phone's gallery app.

Expect: The record saves with one photo; the gallery has an album called **Fieldra** with the photos taken in the app.

Pass ☐ Fail ☐ Notes _____

What this tests: The offline AI that looks at a photo and suggests what it shows (e.g. “plastic”). It must work without internet, and the person’s own answer must always be kept separate from the AI’s guess.

What you get: A suggestion with a percentage (how sure the AI is), made in flight mode.

1. On Wi-Fi: Settings › download the AI (about 250 MB). Then switch on flight mode.
2. Environmental Survey › Waste: photograph a plastic bottle or bag; tap **Suggest classification**.

Expect: A suggestion such as “plastic” with a percentage appears with no internet; your own answer is saved separately from the suggestion.

Pass ☐ Fail ☐ Notes

3. Phone: Terrestrial Survey

Words used in this section

Word	What it means
GCP (ground control point)	A marked target on the ground whose position is measured exactly, so a drone map can be fixed to the real world.
Checkpoint	A target measured the same way but not used to make the map: it is kept back to test how accurate the map is.
RTK	Real-Time Kinematic: a GPS receiver that gets corrections from a base or network and is accurate to about 2 cm, live.
Epoch	One position reading from the receiver, usually one a second. Averaging many epochs gives a steadier position.
Fix (RTK fixed)	The receiver’s best state, accurate to about 2 cm. “Float” is less sure (decimetres).
Pole height / antenna height	The height of the receiver above the mark. It is taken off so the stored height is the ground’s.
Planned position	Where a target should go, from the office plan, before it is measured on site. Stake out leads you there.
GSD	Ground sampling distance: how much ground one photo pixel covers, e.g. 2 cm.
KML	A map file Google Earth and DJI Pilot 2 can open.
TLS station	One place a terrestrial laser scanner (on a tripod) stands to scan.
Setup method	How the scanner’s position is known: over a known point (it stands over a measured mark); resection (it works out its position from 3 or more measured targets it can see); free station with targets (scans are joined through targets two stations both see); cloud-to-cloud (scans are joined by matching their shapes; needs big overlap).
PPK	Post-Processed Kinematic: the drone and a base receiver record raw GPS data; software combines them after the flight to give centimetre positions. Useful where there is no live correction signal.
RINEX	A standard file type for raw GPS data, used for PPK.
UTM	A map grid that gives positions in metres (easting, northing) in zones.

PH-08**Import GCPs from a D-RTK 3 export**

What this tests: Bringing in ground control points (GCPs: marked targets on the ground whose exact position was measured) from a file made by DJI's survey receiver. Importing the same file twice must not create copies.

What you get: Five GCPs on the map, with no duplicates after the second import.

1. Copy `gcps_dr_tk3.csv` to the phone (email, cloud drive or cable).
2. Terrestrial Survey › **Import points** › **Measured with D-RTK 3**; choose the file.
3. Import the same file again.

Expect: 5 added (GCP1–GCP5) the first time; the second time 0 added, 5 updated (no duplicates). They show on the map in Freetown.

Pass ☐ Fail ☐ Notes _____

PH-09**Planned points and stake out**

What this tests: Staking out: walking to a point you planned on a map (e.g. where a target should be laid). The phone points the way and counts down the distance.

What you get: Planned points on the map, and an arrow that leads you to your own point.

1. Import `planned_points.csv` as **Planned positions**. Open P1.
2. Also draw a planned point of your own 20–30 m from you on the Map (Edit › Draw › planned points) and stake it out on foot.

Expect: P1–P3 appear (several kilometres away if you are not in Freetown). For your own point the arrow turns as you turn, the distance counts down, and **At point** shows when you arrive.

Pass ☐ Fail ☐ Notes _____

PH-10**Receiver and GCP observation (simulated)**

What this tests: Measuring a GCP with an RTK receiver (a GPS receiver accurate to about 2 cm). The app collects one position a second (each is an “epoch”) and averages them, which gives a steadier position than a single reading.

What you get: GCP6 saved with an averaged position, its accuracy and two photos.

Needs: Optional: a real Bluetooth RTK receiver instead of the simulator.

1. Settings › turn on the **simulated instrument**.
2. Terrestrial Survey › Connect receiver › Scan; pick the simulated receiver.
3. Survey GCP › source **External RTK**; pole height 2.000; Start observation; wait 30 epochs; take two photos; Save as **GCP6**.

Expect: Fix status, satellites and accuracy show; the epoch count rises once a second; GCP6 is saved with an averaged position and its photos.

Pass ☐ Fail ☐ Notes _____

PH-11**Exports for office software**

What this tests: That the GCPs can be sent to the programs that turn drone photos into maps (DJI Terra, Pix4D, Metashape), each in the layout it expects. UTM is a map grid that gives positions in metres (eastings and northings).

What you get: Five files, one per program or format, with coordinates in the right ranges.

1. Tap each export button: DJI Terra, Pix4D, Agisoft Metashape, Export targets (UTM), Export targets (lat/lon). Send each file to yourself.

Expect: Five files; the UTM file has eastings around 694000–695000 and northings around 934000–935000 for GCP1–GCP5, zone 28N.

Pass ☐ Fail ☐ Notes

PH-12**Flight calculator**

What this tests: The planning calculator for drone flights. GSD (ground sampling distance) is how much ground one pixel of a photo covers: smaller means sharper maps. It depends on the camera and the flying height.

What you get: GSD values that match the known figures for each drone and camera.

1. Choose Matrice 400 + Zenmuse P1 (35 mm) at 100 m. Then Phantom 4 RTK at 100 m.
2. Choose the Zenmuse L3 and change the pulse rate and speed.

Expect: P1 at 100 m: GSD about **1.25 cm**; Phantom 4 at 100 m: about **2.7 cm**. L3: point density goes up with pulse rate and down with speed.

Pass ☐ Fail ☐ Notes

PH-13**Flight log and wind warning**

What this tests: The flight record kept for every drone flight (checks done, take-off and landing times) and the warning when the wind is above what the drone is allowed to fly in.

What you get: A flight log with times a minute apart and a wind warning.

1. New flight log: mission “Test”, Matrice 400, wind **13 m/s**. Tick the 12 checks. Log take-off, wait a minute, Log landing.

Expect: A wind warning shows (limit 12 m/s for the Matrice 400); take-off and landing times are recorded about a minute apart.

Pass ☐ Fail ☐ Notes

PH-14**Mission area KML for DJI Pilot 2**

What this tests: Walking a boundary on foot and saving it as a KML file (a map file Google Earth and DJI’s flight app can open), so the drone can be told where to fly.

What you get: A KML file that opens over the right place in Google Earth.

1. Field Tools › Area walk: tap Add corner at four corners of a yard or car park.
2. Terrestrial Survey: tap the area and export the KML. Open it in Google Earth.

Expect: The KML opens in Google Earth over the right place.

Pass ☐ Fail ☐ Notes

PH-15

Laser scanning station log

What this tests: The log for a terrestrial laser scanner (a scanner on a tripod that measures millions of points around it). Each place it stands is a “station”. The setup method says how the scanner’s position is known (see the words for this section).

What you get: Two stations, S01 and S02, named automatically, with the scanner remembered.

1. New scan station: scanner model, setup **Resection**, tick three targets, resolution, start and finish, a photo. Save. Make a second station.

Expect: The stations are named S01 and S02 by themselves; the scanner model is remembered for the second one.

Pass ☐ Fail ☐ Notes _____

PH-16

PPK job

What this tests: PPK (Post-Processed Kinematic) is a way to get centimetre positions for drone photos after the flight, in the office: the drone and a base receiver on the ground both record raw GPS data, and software combines them. A PPK job keeps everything for one flight in one place.

What you get: A processing sheet listing the base, antenna height, times and files for the office.

Needs: Real RINEX and drone files are optional.

1. Log a flight with positioning **PPK**. Open Terrestrial Survey › PPK jobs › the new job.
2. Choose a base point and antenna height, Log start and Log stop, add any files, then **Share processing sheet**.

Expect: A PPK job is made with the flight; the processing sheet lists the base, antenna, times and files.

Pass ☐ Fail ☐ Notes _____

4. Phone: field modules

Words used in this section

Word	What it means
Point count	Standing still for a set time and counting every animal seen or heard within a distance.
Transect	A line walked while recording what is seen along it.
Quadrat	A small square of ground (e.g. 1 × 1 m) studied in detail, here by photo.
Species richness / Shannon / Simpson / evenness	Numbers for biodiversity: how many species, how varied, and how evenly the individuals are shared between them.
Skip logic	Rules that hide questions that don’t apply, based on earlier answers.
Bearing / azimuth	A direction, as an angle clockwise from north (east is 90°).
Misclosure	In a land survey, how far the walked sides miss the starting beacon. Small is good.
Declination	The angle between magnetic north (compass) and true north (maps).

PH-17**Environmental Survey: every category**

What this tests: That each kind of environmental observation (vegetation, waste, soil, air, noise, general) asks its own questions and saves.

What you get: Six records, one per category.

1. Save one observation for each category: vegetation, waste, soil, air, noise, general.

Expect: Each category shows its own extra boxes (e.g. waste type and amount); six records in Records.

Pass ☐ Fail ☐ Notes _____

PH-18**Biodiversity: sightings and scores**

What this tests: Recording animals or plants seen or heard, and the diversity scores worked out from them. Richness is how many species; Shannon and Simpson are standard numbers for how varied the species are; evenness says how equal their counts are.

What you get: Scores that match the expected values, and a sighting placed 20 m away in the direction you pointed.

1. Quick sighting: **Pied crow**, count 6, seen. Save. **Hooded vulture**, count 3, IUCN CR. **Village weaver**, count 1, heard.
2. Add a distance of 20 m and tap Take bearing on one sighting. Open the scores.

Expect: • Richness **3**, individuals **10**, Shannon **0.898**, Simpson **0.6**, evenness **0.817**. • The sighting with a bearing is drawn 20 m away from you, in that direction. Typing "Pi" suggests Pied crow.

Pass ☐ Fail ☐ Notes _____

PH-19**Biodiversity: point count, transect, quadrat**

What this tests: The three survey methods: a point count (stand still and count for a set time), a transect (walk a line and record along it) and a quadrat (a small square of ground, photographed to estimate plant cover).

What you get: A timed count, a walked line on the map and a plant cover percentage.

1. Point count: 3 minutes, 25 m; let the timer run out. Transect: start, walk 50 m, stop. Quadrat: photograph grass straight down.

Expect: The timer counts down and stops; the transect draws your path; the quadrat gives a cover %.

Pass ☐ Fail ☐ Notes _____

PH-20**Land & Scene Analysis**

What this tests: The AI that splits a photo of a landscape into land cover (trees, buildings, water...), and that the person can correct it.

What you get: A cover list with percentages, and your own class saved separately.

1. Type Land cover: photograph a view with trees and buildings. Analyse photo. Correct the class and save.

Expect: Coloured squares and a cover % list; your class is saved apart from the suggestion.

Pass ☐ Fail ☐ Notes _____

PH-21**Survey plan: study area and visits**

What this tests: Planning fieldwork: drawing the area to cover and the points to visit, then the app ticking points off as records are saved there.

What you get: A plan where the point you stood on becomes “visited”.

1. Biodiversity › Survey plan “Test plan”: Draw area around you; Add points: one where you stand, one 100 m away.
2. Save a sighting where you stand. Then save one outside the area.

Expect: The point you stand on turns **visited**; saving outside the area gives a message.

Pass ☐ Fail ☐ Notes _____

PH-22**Way Back**

What this tests: Finding your way back to where you started (e.g. a vehicle), using GPS and the compass, without signal.

What you get: An arrow and distance back to the start.

1. Mark beginning; walk 100 m with two Save spot; tap the beginning.

Expect: Arrow and distance back; with internet a walking route as well.

Pass ☐ Fail ☐ Notes _____

PH-23**Socio-economic: a form with skip logic**

What this tests: Questionnaires (e.g. household interviews). Skip logic hides questions that don't apply (if consent is “No”, nothing else is asked). Limits stop impossible answers (a household of 60).

What you get: Responses, a CSV table with a column per question, and PDF copies.

1. New form from the household template. Add **household_size** (whole number, 1–50, required) and a choose-several question.
2. New response: answer consent **No**. Then another: consent Yes, household size **60**, then 5.
3. Export CSV; press and hold a response › Share as PDF; on the form, Printable form (PDF).

Expect: With No, the rest is skipped; 60 is refused; the CSV has one column per question name; both PDFs open.

Pass ☐ Fail ☐ Notes _____

PH-24**Photo Gallery**

What this tests: The place where all the project's photos are kept, grouped by what they belong to.

What you get: Photos grouped by category; an added photo under Project photos.

1. Open the gallery; filter by category; Add photo from the library; open a photo and Remove it.

Expect: Photos grouped by category; the added photo is under Project photos; removing asks first.

Pass ☐ Fail ☐ Notes _____

PH-25**Compass and Qibla**

What this tests: Measuring a bearing (the direction to something, as an angle from north, also called an azimuth) and the Qibla direction.

What you get: A saved bearing and the correct Qibla angle for where you are.

1. Compass & Azimuth: point at a far landmark; Record bearing. Compare with a map.
2. Open Qibla.

Expect: Magnetic and true north both shown; in Freetown the Qibla is about **69.5°** (elsewhere compare with a Qibla website).

Pass ☐ Fail ☐ Notes _____

PH-26**Road Survey**

What this tests: Recording a road as a line while driving or walking (even with the screen off), and marking damage such as potholes along it.

What you get: A road line with a pothole marked on it.

1. Road “Test road”, bump indicator on. Start; drive or walk 300 m with the screen off; Record defect here: pothole, severe, with a photo; Pause, Resume, Stop.

Expect: The line keeps recording with the screen off (Android shows a notification); the defect is on the line.

Pass ☐ Fail ☐ Notes _____

PH-27**Land Survey with compass and tape (practical)**

What this tests: Measuring a plot of land with the phone’s compass and a tape: each side’s direction and length. Walking back to the start shows the “misclosure” (how far the sum of the sides misses the start), which tells how careful the measuring was.

What you get: The plot’s area, a precision such as “1 in 1,000”, a PDF report and a beacon list.

1. Peg a rectangle **10 m × 15 m** on open ground (check the diagonals are equal, 18.03 m). Note which way the long side runs.
2. New land survey: tape length 30 m; Use the phone’s declination; Average GPS on beacon 1.
3. For each side: Take bearing (3 s still), type 10 or 15 m, take the back bearing at the next beacon, save. Last side: **Back to B1**.
4. Open the result; Survey report (PDF); Beacon coordinates (CSV).

Expect: Area about **150 m²** (0.015 ha); bearings about 90° apart; precision better than **1 in 300** (1 in 1,000 with care); no side in red unless near metal.

Pass ☐ Fail ☐ Notes _____

PH-28**Water Survey and label scan**

What this tests: Recording a water sample, with its ID read from a printed code on the bottle instead of typed.

What you get: A water sample W-TEST-001 with its readings and photo.

1. New sampling point: **Scan sample label** on the code below; pH **9.0**, turbidity 12, conductivity 800; photo; save.

Expect: The site ID fills in as **W-TEST-001**; the record saves (it is flagged later on the desktop dashboard).

Pass ☐ Fail ☐ Notes _____



Sample label W-TEST-001, for PH-28 and PH-32.

PH-29

Bluetooth water meter

What this tests: Taking readings straight from a Bluetooth water-quality meter into the form, so nothing is typed by hand.

What you get: The form filled from the meter.

Needs: A Bluetooth water meter, or the simulated instrument.

1. Scan for instruments; pick the meter; Use these readings.

Expect: The readings fill the form.

Pass ☐ Fail ☐ Notes _____

5. Phone: Field Tools

Words used in this section

Word	What it means
GNSS	Satellite positioning in general: GPS and its European, Russian and Chinese equivalents.
LiDAR / depth sensing	A sensor that measures distance by light. Only some phones have it (Pro iPhones and iPads, some Android phones).
MGRS	The military grid reference system, another way to write a position.
COGO	Coordinate geometry: positions from bearings and distances.
Resection	Finding where you stand from bearings to known points.
Seed	A number that makes “random” points come out the same each time.

PH-30

GNSS averaging, distance and height

What this tests: Three tools: averaging GPS positions to get a steadier point; measuring a distance by pointing the camera (LiDAR or depth sensing); and working out the height of something from two angles and the distance to it.

What you get: An averaged point, a door height and a building height close to the real ones.

Needs: Distance needs LiDAR (iPhone/iPad Pro) or ARCore depth.

1. GNSS point averaging: stand still 60 s.
2. LiDAR / Depth distance: measure the height of a door (Mark A at the floor, Mark B at the top).
3. Height & slope: a building of known height, from 20 m away.

Expect: The averaged accuracy improves over time; the door measures about 2.0–2.1 m (± 5 cm); the height is within about 5 % of the real one.

Pass ☐ Fail ☐ Notes _____

PH-31**Area walk, barometer, sound, magnetic field**

What this tests: Four sensor tools: area by walking round a shape; height change from air pressure (barometer); noise level; and the magnetic sensor, which jumps near metal.

What you get: An area, a height change of about one floor, a sound level and a magnetic jump.

1. Area walk round the PH-27 rectangle. Barometric elevation: set reference, go up one floor (about 3 m).
2. Sound level for 10 s near a radio; magnetic field: baseline, then pass over a steel tool.

Expect: Area about 150 m² (within 15 % with phone GPS); about +3 m one floor up; the sound level rises with the radio; the magnetic reading jumps over the steel.

Pass ☐ Fail ☐ Notes _____

PH-32**Label scanner**

What this tests: Reading a QR or barcode label with the camera.

What you get: The code's text: W-TEST-001.

1. Point the scanner at the code under PH-28.

Expect: It reads W-TEST-001.

Pass ☐ Fail ☐ Notes _____

PH-33**Coordinate converter**

What this tests: Writing one position in different ways: latitude/longitude, UTM (metres on a map grid), MGRS (the military grid) and national grids. All must describe the same spot.

What you get: The same place in each system, matching the expected values.

1. Type **8.4657, -13.2317**.
2. Under National grid, search "Sierra Leone" and pick a system.
3. Tap Use where I am.

Expect: UTM **28N 694667 936225** (± 1 m); MGRS **28P FQ 94667 36224** (last digit ± 1); DMS 8°27'56.5"N 13°13'54.1"W; national grid values show.

Pass ☐ Fail ☐ Notes _____

PH-34**COGO calculator**

What this tests: COGO (coordinate geometry): working out positions from directions and distances, e.g. a point 100 m east, or where you stand from bearings to three landmarks (resection).

What you get: New points and a distance and bearing that match the expected values.

1. Point at a bearing and distance: from 8.4657, -13.2317, bearing 90° (declination 0), 100 m.
2. Distance and bearing between 8.4657, -13.2317 and 8.4657, -13.2308.
3. Resection: take bearings to three landmarks you can find on the map.

Expect: The new point is about 100 m east (UTM easting +100, northing within 1 m); the distance is about 99 m at bearing about 90°; the resection lands near where you stand and says how well the three bearings agree.

Pass ☐ Fail ☐ Notes _____

PH-35**Sampling points**

What this tests: Making sampling points inside an area for fieldwork: a grid or random points. A “seed” is a number that makes the random points repeatable.

What you get: The same points again with the same seed; new ones with another.

1. Sampling points in the PH-27 area walk: grid at 5 m, seed 7. Then Draw again. Then back to seed 7.

Expect: About 6 points ($150 \text{ m}^2 \div 25 \text{ m}^2$); a new seed moves them; seed 7 gives the same points again; they are saved as planned points, hollow until visited.

Pass ☐ Fail ☐ Notes _____

PH-36**COGO: bearings and distances together**

What this tests: Working out a point from both the directions (bearings) and the distances measured to it from two known points. Using both together is more accurate than bearings alone, and the app says how far each measurement misses.

What you get: The point, a small error ellipse (how sure the position is) and each measurement’s misfit; without distances, a much bigger ellipse.

1. COGO calculator > **A point from bearings and distances**; declination 0; Sighting accuracy 1°, Distance accuracy 0.1 m.
2. A: **8.4657, -13.2317**, bearing **38.5°**, distance **70.7 m**. B: **8.4657, -13.2308**, bearing **315°**, distance **78.0 m**.
3. Then clear both distances and calculate again.

Expect: With distances: **8.46620, -13.23130** (within 5 cm), a 95 % ellipse of a few centimetres, and **Measurements miss by** under 0.2° and 1 cm for each. With bearings only: the same point within about a metre, but the ellipse grows to about **±3.4 m × ±3.0 m**. Save as a point keeps the ellipse in its details.

Pass ☐ Fail ☐ Notes _____

PH-37**Sample size for a target precision**

What this tests: How many sampling points are needed for the average to be as precise as you want (e.g. pH to ±0.2), given how much the value varies. More confidence needs more points.

What you get: The number of points for 95, 99 and 90 % confidence, put into the design.

1. Sampling points > **How many points?**: spread (standard deviation) **0.6**, precision **0.2**, confidence 95 %. Then 99 % and 90 %.
2. Back to 95 %; tap **Use 35 points**.

Expect: **35** points at 95 %, **60** at 99 %, **25** at 90 %; Use puts 35 into the number of points of the design.

Pass ☐ Fail ☐ Notes _____

6. Phone: map, layouts and export

Words used in this section

Word	What it means
Buffer	The area within a set distance of a place or feature.
Snap	A tap close to an existing corner jumps exactly onto it.
GeoJSON / KML / CSV	Common file types: map data (GeoJSON, KML) and spreadsheet tables (CSV).

PH-38 Map navigation and layers

What this tests: Moving round the map and choosing what it shows: going to a coordinate, saved views (bookmarks), layers on and off, colouring by a value and filtering by date.

What you get: The map at the right place, with the legend and time filter working.

1. My location; Full extent; Go to XY **28N 694667 936224**; Bookmarks: save “Freetown” and come back to it.
2. Layers: switch layers on and off; Streets / Satellite; Colour by: water condition; North up; Time: last 7 days.

Expect: Go to XY marks the spot in Freetown; the bookmark returns to the same view; the legend shows each colour; older records hide with the time filter.

Pass ☐ Fail ☐ Notes

PH-39 Terrain, measure and buffer

What this tests: Ground shape on the map (relief, contour lines, 3D), measuring along the ground, and a buffer: everything within a set distance of a place.

What you get: A 3D view, a height profile and a list of records within 100 m.

Needs: Terrain needs internet.

1. Relief, Contours and 3D (orbit ball: drag, side handle, top handle, double-tap). Profile along three points.
2. Distance along a path; Area of a yard and Save; Buffer 100 m around you.

Expect: 3D with relief; the profile shows lowest, highest, climb and steepest part; the buffer card counts records inside and lists them.

Pass ☐ Fail ☐ Notes

PH-40 Snap and drawing

What this tests: Snapping: when drawing, a tap close to an existing corner jumps exactly onto it, so shapes join without gaps.

What you get: A new area that shares a corner exactly.

1. Edit > Draw a study area with Snap on, starting on a corner of an existing area.

Expect: The tap jumps onto the corner.

Pass ☐ Fail ☐ Notes

PH-41**Layout PDF and image**

What this tests: Making a printable map page with title, legend, north arrow and scale bar.

What you get: A one-page PDF and a picture of the map.

1. Layout: frame the records, landscape, Next; switch the legend off and on; Export PDF (A4) and Export image. Save image from the map.

Expect: A one-page PDF with title, legend, north arrow, scale bar and date; the image matches the screen.

Pass ☐ Fail ☐ Notes

PH-42**Offline map**

What this tests: Downloading the map for an area so it still shows where there is no signal.

What you get: A map that works in flight mode.

1. On Wi-Fi: Offline map › Download this area. Switch on flight mode and pan the area.

Expect: The map still shows with no signal.

Pass ☐ Fail ☐ Notes

PH-43**Records and export**

What this tests: The list of everything saved, and getting it out of the phone in common file types: GeoJSON and KML (map files), CSV (a spreadsheet table) and ZIP (all together, with photos).

What you get: Files that open in a spreadsheet and in Google Earth.

1. Records: filter by type; open one; delete one (press and hold).
2. Export GeoJSON, CSV, KML and ZIP. Open the CSV in a spreadsheet and the KML in Google Earth.

Expect: Each file opens; the ZIP holds all of them plus the photos; records show PENDING UPLOAD until synced.

Pass ☐ Fail ☐ Notes

7. Desktop: install, screen and projects

Install Fieldra Desktop (store or the /beta page) on a Mac or Windows PC.

DT-01 Install and first start

What this tests: That Fieldra Desktop installs and opens, and is allowed to find phones on the network.

What you get: The main window with ribbon, layer list, map and status bar.

1. Install; on a Mac Allow it to find devices on the local network; on Windows allow it through the firewall on private networks.
2. Fill in the welcome survey.

Expect: The main screen opens: title bar, ribbon, Contents, map, right pane, status bar with scale and coordinates.

Pass ☐ Fail ☐ Notes

DT-02 Projects and keyboard shortcuts

What this tests: Making a project on the computer and the keys that open each tab or tool.

What you get: Each shortcut opening its tab or tool.

1. File › New Project (Cmd/Ctrl+N): **Test Site**. Press Cmd/Ctrl+1 ... 5. Press H, S, M, T.

Expect: Each shortcut opens its tab or tool (Map, Edit, Analysis, Drone, Share).

Pass ☐ Fail ☐ Notes

DT-03 Command search, sample project, jobs, autosave and history

What this tests: Finding tools by everyday words, the ready-made sample project, stopping a long job, and that work is saved by itself with earlier versions you can go back to.

What you get: Edits kept after a restart, and an earlier version put back.

1. Cmd/Ctrl+K: type “watershed”, then “interpolate”. Open the sample project from Contents. Start Empirical Bayesian kriging on the water samples and press Stop in the status bar.
2. Edit a layer; wait 2 minutes; edit again; quit Fieldra and open it again. Right-click the layer › History: put back as it was...; Put back the older version.
3. Click Light / Full in the status bar.

Expect: Search finds Hydrology and Interpolate by everyday words; the sample project opens styled with its images; the stopped job ends quietly. After restarting, the edits are there; History lists the earlier version and Put back restores it (the state before is listed too). Light mode makes the map less sharp on high-density screens and smoother on slow computers; “auto” follows the computer.

Pass ☐ Fail ☐ Notes

DT-04**Project package**

What this tests: Saving a whole project as one file (.fieldra) to back it up or pass it on, and opening it again.

What you get: A .fieldra file that brings the whole project back.

1. After DT-05 has added a few layers, Share › Export package. Delete the project, then Share › Open package.

Expect: A .fieldra file; opening it brings back the records, photos, layers and drone files.

Pass ☐ Fail ☐ Notes

8. Phone and desktop together

PD-01**Connect a phone and receive**

What this tests: Pairing a phone with the computer (by scanning a code) so the phone's records copy to the computer over Wi-Fi, with no cloud account.

What you get: The phone's records on the computer, marked SYNCED on the phone.

1. Desktop: Share › Connect phone. Phone: Settings › Fieldra Desktop › Connect; scan the code.

Expect: Everything on the phone arrives; a card shows what is coming in; on the phone, records change to SYNCED. A new record saved later arrives by itself.

Pass ☐ Fail ☐ Notes

PD-02**Survey data on the desktop**

What this tests: That records from the phone keep all their details, colours and photos on the computer.

What you get: Full details and photos for each record.

1. Click a water sample and a sighting in Survey data. Map › Photos.

Expect: All details and photos show; the phone's colours are used; photos are grouped by category.

Pass ☐ Fail ☐ Notes

PD-03**Nearby Devices: desktop to phone with layers and an orthophoto**

What this tests: Sending a project from the computer to a phone, including map layers and an orthophoto (an aerial photo corrected so it lines up with a map).

What you get: The layers and orthophoto on the phone, usable offline.

1. Add districts.geojson and ortho_2026-01.tif to Test Site (as in DT-05 and DR-04). Style districts by name.
2. Desktop: Share › Nearby devices › send Test Site to the phone (phone: Nearby Devices open). Keep the orthophoto ticked.

Expect: On the phone, the districts show in their desktop colours; the orthophoto is under Map › Offline maps, switched on, and shows with no signal.

Pass ☐ Fail ☐ Notes

PD-04**Interrupted transfer and deletions**

What this tests: That a transfer cut off half way can carry on where it stopped, and that deletions are passed on too.

What you get: A finished transfer without starting again; the deleted record gone on the phone.

1. Start sending again with a large orthophoto; turn the phone's Wi-Fi off halfway; on again; tap **Continue** under Unfinished.
2. Delete a record on the desktop and send again.

Expect: The transfer carries on where it stopped; the deleted record disappears on the phone; only changes are sent the second time.

Pass ☐ Fail ☐ Notes _____

PD-05**Phone to phone**

What this tests: Sending a project straight from one phone to another.

What you get: The project, with photos, on the second phone.

Needs: A second phone.

1. Two phones, Nearby Devices open on both; rename one; send a project; try Add by address.

Expect: The project arrives with photos; Recent transfers lists it with Open.

Pass ☐ Fail ☐ Notes _____

PD-06**Edit desktop layers on the phone and sync back**

What this tests: Editing office map layers in the field (moving, adding and deleting points), sending the changes back, and what happens when the same feature was changed in both places.

What you get: The computer's layer updated with the phone's edits, and the old version kept in History.

Needs: A paired phone on the same Wi-Fi.

1. Desktop: add valves .geojson to Test Site; Layer properties › Fields and rules: valve is text, required. Send Test Site to the phone.
2. Phone: Map › Edit › Desktop layers › valves. Open V2; change valve to "V2-checked"; Save. Open V1; Move to where I stand. Add a point where I stand; name it V4; Delete V3.
3. Meanwhile on the desktop: change V1's valve to "V1-desk". Then let the phone sync (same Wi-Fi), or Send now.

Expect: The phone shows "4 changes waiting" until it syncs, then none. On the desktop a message says the phone edited valves: V2 is "V2-checked", V4 is new, V3 is gone, and V1 is at the phone's position with the phone's values (both edited V1: the phone's version of a feature it changed wins; desktop edits to features the phone didn't touch stay). Right-click valves › History lists the version before the phone's edits; Put back restores it.

Pass ☐ Fail ☐ Notes _____

PD-07**Reshape a desktop area on the phone**

What this tests: Changing the shape of an office area on the phone, corner by corner, and sending it back.

What you get: The new shape on the computer, with the old one in History.

Needs: A paired phone.

1. Desktop: add `districts.geojson` to Test Site and send it. Phone: tap one of its areas › **Reshape on the map.**
2. Step to corner 2 with ☐; tap the map a little outside to move it; Add a corner after it; Remove corner 4; Save shape. Sync.

Expect: The numbered corners follow the steps; Undo all brings the shape back; after Save the area is redrawn, and after sync the desktop shows the new shape (History holds the old one).

Pass ☐ Fail ☐ Notes

PD-08**Web layers and downloaded features on the phone**

What this tests: Showing online map layers on the phone (tiles: pictures of the map) and downloading online features (points, lines, areas) to edit offline.

What you get: Tiles under the records and an editable downloaded layer.

Needs: Internet on the phone.

1. Phone: Map › Layer › **Web layers.** Add `https://tile.openstreetmap.org/{z}/{x}/{y}.png`; set it half see-through; switch it off and on.
2. Download features: `https://sampleserver6.arcgisonline.com/arcgis/rest/services/Wildfire/FeatureServer/0`; open one, change a value, Save.

Expect: The tiles draw under the records at the chosen see-through; the features come in as a phone layer that can be edited, and the edit stays on the phone.

Pass ☐ Fail ☐ Notes

PD-09**Datum grid file sent to the phone**

What this tests: Some national coordinate systems are converted exactly only with a grid file from the survey authority. The desktop sends that file to phones with the project, so the phone's converter becomes exact too.

What you get: The phone's values move to the exact ones and say so; the grid isn't sent twice.

Needs: A paired phone.

1. Phone first: Field Tools › Coordinate converter: **8.4657, -13.2317**; National grid: **EPSG:4175** (Sierra Leone 1968). Note the values and the note under them.
2. Desktop: Share › Datums & heights › Grid shift: Sierra Leone 1968 with `sl1968_shift.gsb`. Send Test Site to the phone (Nearby devices).
3. Phone: the converter again, same position and system. Then send the project a second time.

Expect: Before: about **8.465599, -13.231552**, marked approximate. After: **8.465422, -13.231144** (1" south and 2" east of WGS 84; about 49 m from before) and "Using the grid file from Fieldra Desktop: exact." The desktop shows the same values. The second send doesn't send the grid again.

Pass ☐ Fail ☐ Notes

PD-10**Save phone edits back to a feature service**

What this tests: A layer downloaded from an online map service (ArcGIS, WFS, OGC API) can be edited on the phone and saved back to the service, with a check that nobody else changed the same features meanwhile.

What you get: Edits saved to the service; a clear choice when someone else changed the same feature.

Needs: Internet on the phone (the sample service is public and may be reset at any time).

1. Phone: Web layers › Download features: <https://sampleserver6.arcgisonline.com/arcgis/rest/services/Wildfire/FeatureServer/0>. Open the layer.
2. Change a value of one feature and move another; the **Web service** section counts the changes. Tap **Save 2 changes to the service**.
3. Change the same feature again on the phone; meanwhile change it on the service too (Fieldra Desktop: add the same layer, edit that feature, Save). Save on the phone.
4. Tap **Get the latest from the service**.

Expect: The first save goes through and the count drops to none. The second finds the conflict and asks: **Keep theirs** takes the service's version, **Overwrite with mine** sends the phone's. Get the latest brings in other people's changes. A service that refuses edits gives its own message.

Pass ☐ Fail ☐ Notes _____

PD-11**Team server**

What this tests: A small program a team can run to pass records from phones to computers when they are not on the same Wi-Fi.

What you get: A record that reaches the computer without sharing a network.

Needs: The team server program; a computer both can reach (same network or the internet).

1. On any computer: Fieldra-TeamServer -code 482913 (it prints its address). Phone: Settings › Fieldra Desktop › Connect: type the address and 482913. Save a record.
2. Desktop: Share › **Team server**: the same address and code, Connect; Pull now. Try a wrong code on another phone ten times.

Expect: The record and its photo arrive on the desktop without being on the same Wi-Fi as the phone; the dialog shows when it last pulled and how many items came. After ten wrong codes that phone is refused for a while (too many attempts).

Pass ☐ Fail ☐ Notes _____

PD-12**Open files, share a layer, record a track**

What this tests: Opening map files on the phone, sending a layer as a file, recording a walk as a track, and searching for a place by name.

What you get: Three opened layers, a shared KML, a saved track and the place found.

Needs: A paired phone; internet for the place search.

1. Copy `boreholes.kml`, `walk_track.gpx` and `water_samples.csv` to the phone. Map › Edit › Data layers › **Open a file**: pick all three.
2. Open `boreholes` › Share as a file › KML; send it to yourself. Field Tools › **Record a track**: on foot, Start recording, walk about 100 m, Pause, walk 20 m, Resume, walk back; Save as “Test walk”.
3. Map › Go to XY: type **Connaught Hospital**, Search, pick it.

Expect: Three layers appear: `boreholes`, `walk_track` (a line) and `water_samples` with 40 points and their fields. The shared KML opens in Google Earth. The track shows its length and time while recording; saved, it is in the Tracks layer with `length_m`, `duration_min` and `speed_kmh`, in two parts. After the next sync the layers and the track are on Fieldra Desktop in the project. The place search lists the hospital in Freetown and the map goes there.

Pass ☐ Fail ☐ Notes

PD-13**Split an area and overlays on the phone**

What this tests: Cutting an area in two along a line between two of its corners, and simple overlays in the field: keeping what is inside areas, or counting points in each area.

What you get: Two areas with the same details; a clipped layer; each district with its number of samples.

Needs: A paired phone.

1. With `districts.geojson` and `water_samples.csv` on the phone (from the tests above): `districts` › tap North-West › Reshape on the map.
2. Choose corner 1, **Split from here**; choose corner 3, **Split to here**; Save shape. Then try two corners next to each other.
3. `water_samples` › **Overlay** › Clip to an area › `districts`. Then `districts` › Overlay › **Count another layer in each of these areas** › `water_samples`, field `ph`.

Expect: North-West becomes two triangles with the same name and population (the second a new feature); corners next to each other are refused with a message. The clip keeps the samples inside the districts; the count gives each district its samples — North-West 11 (over its two halves), North-East 8, South-West 10, South-East 11 — with `sum_ph` and `mean_ph`. The new layers go to the desktop with the next sync.

Pass ☐ Fail ☐ Notes

PD-14**Land survey on the desktop**

What this tests: That a land survey done on the phone shows the same results on the computer and can be adjusted there.

What you get: The same area and precision, a report and a beacon list.

1. After PH-27 has synced: Map › Land survey; open the parcel. Office settings: type a surveyed position for beacon 1. Survey report; Beacons CSV; Add to map.

Expect: The same area and precision as the phone; the beacon schedule in lat/lon and UTM; moving beacon 1 moves the whole plot.

Pass ☐ Fail ☐ Notes

PD-15 **Forms from the desktop to the phone**

What this tests: Bringing in a questionnaire made in KoboToolbox or ODK (an XLSForm spreadsheet), filling it in on the phone, and the reports on the computer.

What you get: The form working on the phone and its answers in the reports.

1. Map › Survey forms › New form › Import XLSForm: household_water_xlsform.xlsx. Send the project to the phone; answer twice on the phone; sync.
2. On the desktop: reports as PDF, Word and CSV.

Expect: The form arrives with its questions numbered and the skip rules working; Fieldra lists **location** (GPS) and **photo** as not brought over; the two responses are in the summaries.

Pass ☐ Fail ☐ Notes _____

9. Desktop: adding and exporting data

Words used in this section

Word	What it means
Vector / raster	Vector data is points, lines and areas; raster data is an image of cells (photos, elevation models).
Coordinate system (CRS)	The way positions are written; the same place has different numbers in each (e.g. latitude/longitude or UTM).
Geodatabase (.gdb)	Esri's file format for many layers and tables together.
Domain / subtype	Rules on a field: a pick-list or a number range; subtypes group features with their own rules.
Geoid	The difference between GPS heights (above a mathematical earth) and heights above sea level.
WMS / WFS / OGC API	Standard web services for map pictures (WMS) and for features you can query and edit (WFS, OGC API).
PostGIS / SQL Server	Databases that store map data for a team.

DT-05 Add the vector files

What this tests: Opening the common map file types. Vector data is points, lines and areas (as opposed to images).

What you get: Each layer with the right number of features in Freetown.

1. Add data (Cmd/Ctrl+O) or drag in: `water_samples.csv`, `districts.geojson`, `study_area.geojson`, `roads.geojson`, `clinics.geojson`, `households.geojson`, `boreholes.kml`, `walk_track.gpx`.

Expect: `water_samples` 40 points, `districts` 4, `roads` 60 lines, `clinics` 2, `households` 30, `boreholes` 3, one track; all in Freetown.

Pass ☐ Fail ☐ Notes _____

DT-06 Tables without coordinates

What this tests: Opening plain tables that have no position, to join to map layers later.

What you get: Two tables with 38 and 13 rows.

1. Add `lab_results.csv` and `site_visits.csv`.

Expect: They open as tables (no points): 38 and 13 rows.

Pass ☐ Fail ☐ Notes _____

DT-07**CAD drawing**

What this tests: Opening a CAD drawing (DXF, from AutoCAD and similar) in the right place on the map.

What you get: The drawing's layers, in their colours, with a boundary of the right area.

1. Add `site_plan.dxf` with `site_plan.prj` beside it.

Expect: Layers for BOUNDARY (an area of **4,800 m²**), ROADS (2 lines), BOREHOLES (1 point) and TEXT ("BH1 depth 32 m"), in their CAD colours.

Pass ☐ Fail ☐ Notes _____

DT-08**Export for ArcGIS and round trip**

What this tests: Saving layers in the formats ArcGIS and other GIS programs use, then opening them again to check nothing was lost or moved.

What you get: Layers that come back with the same fields and positions.

1. Export coordinates: **WGS 84 / UTM 28N**. Share › Export for ArcGIS (GeoPackage); Share › Export .gdb; Export DXF; right-click `water_samples` › Export Shapefile.
2. Add the GeoPackage, the .gdb and the Shapefile back into a new project.

Expect: Every layer comes back with its fields; the Shapefile's .prj says UTM zone 28N; positions match the originals.

Pass ☐ Fail ☐ Notes _____

DT-09**File geodatabase: domains, subtypes and spatial index**

What this tests: That rules on fields (a pick-list, or a number range) survive being saved to an Esri file geodatabase (.gdb).

What you get: The same rules on the re-opened layer.

1. An editable copy of `water_samples`: Layer properties › Fields and rules: condition one of clear / turbid / polluted; ph a number from 0 to 14. Save.
2. Right-click it › Export › File geodatabase. Add the .gdb back into a new project and open its Layer properties.

Expect: The re-opened layer has the same rules: condition is a pick-list of the three values, ph is 0 to 14. In ArcGIS Pro (if you have it): the feature class has domains **..._condition** (coded) and **..._ph** (range) in the field properties, and a spatial index.

Pass ☐ Fail ☐ Notes _____

DT-10**National grids**

What this tests: Finding a country's own coordinate system and the shift between it and GPS coordinates (WGS 84).

What you get: The national systems listed first and the shift shown.

1. Export coordinates › National grid or other system: search "Sierra Leone", then "EPSG:32628". Open Change datum shift.

Expect: Systems for Sierra Leone are listed first; EPSG:32628 is WGS 84 / UTM zone 28N; the datum shift to WGS 84 shows.

Pass ☐ Fail ☐ Notes _____

DT-11**Site grid**

What this tests: A site grid is a local grid set up on a construction or mine site. Fieldra works out how it is turned and scaled compared with UTM from points known in both.

What you get: The rotation, scale and fit at each point.

1. Share › Datums & heights › Site grid: enter the four stations of `site_grid_points.csv` (site E, N and UTM E, N). Save it as “Test grid”.

Expect: Rotation **12°**, scale **1.0000**, misfit under 1 cm at every station; Test grid can be chosen in Go to XY and exports.

Pass ☐ Fail ☐ Notes _____

DT-12**Geoid and heights**

What this tests: GPS gives heights above a mathematical shape of the earth (the ellipsoid); maps use heights above sea level. The geoid is the difference between the two. This test adds sea-level heights.

What you get: A new height field H on the points.

Needs: Internet for the download.

1. Geoid: download EGM2008 (once). Heights: add H to the boreholes or to the GCPs from the phone.

Expect: An H field is added; $H = h - N$, so H differs from h by the same amount (the geoid height) at points close together.

Pass ☐ Fail ☐ Notes _____

DT-13**Web map layers**

What this tests: Showing online background maps (WMS and XYZ tile services) and keeping what was already seen for offline use.

What you get: Online maps under your data, still there offline.

Needs: Internet. Public test services can change; use any WMS you have if one is down.

1. Map › Web layer: WMS <https://ows.terrestris.de/osm/service>, layer OSM-WMS.
XYZ: <https://tile.openstreetmap.org/{z}/{x}/{y}.png>.
2. Pan an area; go offline; pan the same area.

Expect: Both draw under your data; areas already seen still show offline.

Pass ☐ Fail ☐ Notes _____

DT-14 Feature services (WFS, OGC API)

What this tests: Bringing in online features (not just pictures) from standard web services, editing them and refreshing.

What you get: A linked layer whose edits are counted and can be refreshed.

Needs: Internet. For saving, a service of your own that allows editing (e.g. GeoServer WFS-T).

1. Web layer › Features: OGC API <https://demo.pygeoapi.io/stable>, a collection such as lakes; Only the map view.
2. Edit one feature; note the unsaved-edit count; Refresh; Unlink.

Expect: The features come in as a linked layer; the bar counts your edits; Refresh brings back the service's version; Unlink keeps a plain layer. On a service that allows editing, Save sends the edit and Refresh shows it saved.

Pass ☐ Fail ☐ Notes

DT-15 ArcGIS Online and Enterprise layers

What this tests: Using Esri's online layers: bringing them in, editing and saving back.

What you get: Features from the service, an edit saved, and a background map image.

Needs: Internet. Esri's public sample services reset their data now and then; any FeatureServer you can edit works.

1. Web layer › Features: <https://sampleserver6.arcgisonline.com/arcgis/rest/services/Wildfire/FeatureServer>. Pick a layer (it says "editable"); Only the map view off; Add.
2. Edit one feature and Save; Refresh. Then Web layer › Map images: <https://sampleserver6.arcgisonline.com/arcgis/rest/services/USA/MapServer>.

Expect: The layers and tables of the service are listed; the features come in whole with their dates; the edit is saved (Refresh shows it). The MapServer draws as a background image.

Pass ☐ Fail ☐ Notes

DT-16 Related tables and attachments

What this tests: Linking a table to a layer (one sample has several visits) and attaching files to a feature.

What you get: Each sample showing its visits; a PDF attached.

1. water_samples › Layer properties › Related: link to site_visits by **site_id**. Click W01, then W02.
2. On W01, + to add a visit. In its details, Attach a PDF.

Expect: W01 lists 3 visits, W02 2; the new visit has site_id W01 filled in; the attachment opens in its usual program.

Pass ☐ Fail ☐ Notes

DT-17 Relationships in a file geodatabase

What this tests: That the link between a layer and a table is kept when saved to a .gdb and opened again.

What you get: The link working again after re-opening.

1. With the water_samples ↔ site_visits relationship from the last test: Share › Export .gdb. Remove both layers; Share › Open .gdb on the saved file.

Expect: site_visits comes back as a table (no shapes), and the relationship is set up again by itself: W01 lists its 3 visits. ArcGIS Pro (if you have it) shows the relationship class.

Pass ☐ Fail ☐ Notes

DT-18**netCDF variables and curved .gdb edges**

What this tests: netCDF is a file type for climate and ocean grids with several variables. Also, curved boundaries in a .gdb must stay curved, not cut into straight lines.

What you get: Two images from the netCDF file and a curved plot with the right area.

Needs: For netCDF, a GDAL on the computer (comes with QGIS).

1. Add `climate.nc`: tick both variables. Then Share › Open .gdb on `curved_plot.gdb.zip` and measure the plot's area.

Expect: The file lists **rain_mm** and **temp_c**; each becomes an image (rain 200 to 260 mm west to east, temperature 26 to 28 °C south to north). The plot's north-east corner is a smooth quarter circle and its area about **9,914 m²** (cut straight it would be 9,800 m²).

Pass ☐ Fail ☐ Notes

DT-19**Sign-ins in the keychain and edit conflicts**

What this tests: That passwords for online services are kept safely by the computer, and what happens when two people change the same feature.

What you get: The password in the keychain; a choice to overwrite or keep the other change.

Needs: A service that allows editing; two computers for the conflict.

1. Web layer › Sign-in for a service of your own: add a user and password. Open Keychain Access (Mac) or Credential Manager (Windows) and look for Fieldra.
2. Download a feature layer you can edit on two computers. Change the same feature on both; Save on the first, then on the second.

Expect: The sign-in window says the password is kept in the keychain, and the entry is there. On the second computer Fieldra lists the feature changed by someone else and asks: overwrite or keep theirs.

Pass ☐ Fail ☐ Notes

DT-20**PostGIS: export, edit live, history and review**

What this tests: Working with a PostgreSQL/PostGIS database (a shared database for map data): saving to it, editing it directly, seeing who changed what, and proposing changes for review.

What you get: A table in the database with a full change history.

Needs: PostgreSQL with PostGIS (Docker is the easiest way).

1. Start a database, e.g. `docker run -d -p 5432:5432 -e POSTGRES_PASSWORD=fieldra postgres/postgis`.
2. Share › Databases (PostgreSQL / PostGIS): connect (user postgres). Export `water_samples` as a new table; Add it back with **Edit live**.
3. Change W05's pH; Save. History: type your name; see the change; Restore it. Make another edit and **Propose** it; approve it in Proposals.
4. In History tick Keep changes made by any program; change a row in another tool (psql or QGIS); Refresh.

Expect: The table has 40 rows and a spatial index; the history lists who changed what; Restore puts the old pH back; the proposal is saved under your name when approved; the other program's change is in the history too.

Pass ☐ Fail ☐ Notes

DT-21**SQL Server: export, add and edit live**

What this tests: The same with Microsoft SQL Server, including not overwriting a change someone made in the meantime.

What you get: A table in SQL Server; changes made elsewhere kept.

Needs: SQL Server 2016 or later (Docker is the easiest way).

1. Start a server, e.g. `docker run -d -p 1433:1433 -e ACCEPT_EULA=Y -e MSSQL_SA_PASSWORD=Fieldra!2026 mcr.microsoft.com/mssql/server:2022-latest`, and make a database `gis`.
2. Share › Databases: Database type SQL Server, user `sa`, tick Trust the server's certificate; Connect. Export `water_samples` with the export system set to UTM 28N; then Add it, and Add it again with **Edit live**.
3. Move a point and change a value; Save. In SQL (e.g. Azure Data Studio) change another row's name; then change the same row in Fieldra and Save.

Expect: The table is listed as point · EPSG:32628 with 40 rows; added, the points sit exactly on the originals. Your save goes in; the row changed in SQL is not overwritten — it is listed, and Refresh shows the SQL change. History and Propose are not shown for SQL Server.

Pass ☐ Fail ☐ Notes _____

DT-22**Go to XY, bookmarks and time**

What this tests: Going to a position typed in different ways, saving views, and playing records through time.

What you get: All positions at the same place; the time slider stepping through 10 days.

1. Go to XY: **28N 694667 936224**, then **28P FQ 94667 36224**, then **8°27'56.5"N 13°13'54.1"W**; Also show a national grid.
2. Bookmarks: save the site view. Map › Time with `water_samples`: press play.

Expect: All three go to the same place; the slider steps through **10** days with **4** samples each.

Pass ☐ Fail ☐ Notes _____

10. Desktop: editing and symbology

Words used in this section

Word	What it means
Symbology	How features are drawn: colours, sizes and symbols.
Annotation	Map text stored on its own so it can be moved by hand.
Topology	How shapes fit together: no overlaps, gaps or crossings.

DT-23 Draw, snap, trace, undo

What this tests: Drawing new features, snapping to existing ones, tracing along a line, and undoing.

What you get: Shapes that join exactly and an undo that works.

1. New layer; draw a point, a line, an area, a rectangle, a circle, a freehand shape. Snap a new area to a district corner. Trace along Main Road between two junctions.
2. Undo and redo (Cmd/Ctrl+Z, Shift+Cmd/Ctrl+Z).

Expect: Snapped corners sit exactly on the district corner; the trace follows the road; undo removes the last map change.

Pass ☐ Fail ☐ Notes

DT-24 COGO from a deed

What this tests: Drawing a plot from a deed (a legal description: a list of bearings and distances), and the misclosure that shows how well it closes.

What you get: The plot with its length, misclosure and area.

1. Edit > COGO: start **28N 695600 935600**; enter the five legs of deed_traverse.txt.

Expect: Length **244.88 m**; misclosure **0.006 m** (1 in about 43,299); the closed area is **4,010.7 m²** ($\pm 0.5 \text{ m}^2$).

Pass ☐ Fail ☐ Notes

DT-25 Edit tools

What this tests: The editing tools: merge, copy parallel, move, rotate, generalise (simplify), split, extend and trim.

What you get: One merged road of about 2 km; every tool undoable.

1. Select the five Main Road lines (By attributes: [name] = 'Main Road') > **Merge**.
2. Copy parallel 10 m; Move 50 m east; Rotate 10°; Generalise 5 m; Split an area with a line; Extend a line to another; Trim.

Expect: 5 lines become one of about **2,000 m**; each tool is one undo step.

Pass ☐ Fail ☐ Notes

DT-26**Style by attribute and rules**

What this tests: Colouring features by their values: by category, by graduated classes (ranges of a number) and by rules you write.

What you get: The right number of features in each colour.

1. water_samples › Symbology › Style by **condition** (categories). Then graduated colours on **ph** (natural breaks, 5 classes). Then Rules: [ph] > 8.5 red, [condition] = 'polluted' orange; untick Show everything else.

Expect: Categories: clear **16**, turbid **14**, polluted **10**; graduated classes from 6.56 to 8.91; rules show **4** red points (the rest of the shown points orange).

Pass ☐ Fail ☐ Notes _____

DT-27**Labels, label classes and annotation**

What this tests: Writing text on the map from the data, styling some labels differently, and turning labels into annotation (text you can move by hand).

What you get: Labels such as “W05 (7.12)” that can be moved one by one.

1. Label with the expression [site_id] + ' (' + [ph] + ')'; best placement; halo.
2. Label classes: polluted ones bold red. Convert labels to annotation; move one.

Expect: Labels such as “W05 (7.12)”; polluted labels differ; after conversion each label moves on its own.

Pass ☐ Fail ☐ Notes _____

DT-28**Field rules, subtypes and Check data**

What this tests: Rules on what a field may hold, and finding the features that break them.

What you get: Exactly the two broken features selected.

1. Layer properties of an editable copy of water_samples: ph is a number 0–14, condition is one of clear / turbid / polluted.
2. Set one pH to 15 and one condition to “muddy”; Check data.

Expect: Exactly those two features are selected; condition is a drop-down while editing.

Pass ☐ Fail ☐ Notes _____

DT-29**Calculated fields**

What this tests: Fields that fill themselves from a formula (e.g. area in hectares) and update when the shape changes.

What you get: Areas per field that change by themselves after a reshape.

1. An editable copy of fields (from the sample project or the pack): Layer properties › **Calculated** › + Area in hectares; + Your own: size = if([area_ha] > 2, 'large', 'small'). Save.
2. Reshape the Pond field to about half its size.

Expect: area_ha appears on every field: Maize **3.96**, Cassava **1.8**, Bare soil **2.64**, Pond **1.1**, Village **1.54** ha (Maize and Bare soil “large”). After the reshape, Pond’s area_ha drops to about half by itself; the other fields don’t change.

Pass ☐ Fail ☐ Notes _____

DT-30**Topology: find and fix**

What this tests: Topology means how shapes fit together: no overlaps, no gaps, no shapes crossing themselves, no copies. This test finds and fixes each problem.

What you get: Each problem found, drawn and fixed.

1. Add `parcels_topology.geojson`. Layer properties › Topology: no overlaps, no gaps, must not cross itself, no duplicates. Check data. Then Fix.

Expect: Errors: an **overlap** of P2 and P3 (5 × 40 m), a **gap** between P4 and P5 (2 m wide), P6 **crosses itself**, P7 **duplicates** P1; drawn in "Topology errors". Fix removes each, one undo step each.

Pass ☐ Fail ☐ Notes _____

DT-31**Topology between layers**

What this tests: Rules between two layers, e.g. every pipe end must have a valve, and every valve must sit on a pipe.

What you get: The right number of problems of each kind.

1. Add `pipes.geojson` and `valves.geojson`. Layer properties of pipes › Topology: Lines must not overlap; Lines must not cross; Line ends must have a point (valves, within 0.5 m).
2. Layer properties of valves › Topology: Points must lie on a line (pipes, within 0.5 m). Check data on both.

Expect: Pipes: **P1 and P3** run along each other for **120 m**; **2** crossings (P4 across P1 and P3); **6** line ends without a valve. Valves: **V2** is **3.0 m** from the nearest line.

Pass ☐ Fail ☐ Notes _____

DT-32**Drawing options**

What this tests: Ways to draw layers: see-through, hatched, dashed, your own symbol, only at some scales, heat maps and clusters.

What you get: Each option showing as described.

1. Districts: 50 % transparency, hatched fill; roads dashed; boreholes as your own PNG; show `water_samples` only from 1:25,000 in; heat map of `water_samples` weighted by `turbidity_ntu`; then clusters.

Expect: Each option shows; the heat is strongest along Main Road; clusters show counts and split as you zoom in.

Pass ☐ Fail ☐ Notes _____

DT-33**3D**

What this tests: 3D views: raising areas by a value, terrain, buildings, a globe, and turning the view.

What you get: Districts raised by population in 3D.

1. Districts: raise by population. Terrain, Buildings, Globe; orbit ball; Flat view.

Expect: North-East is tallest (22,150 people), South-West lowest; double-click on the orbit ball goes back to flat, north up.

Pass ☐ Fail ☐ Notes _____

DT-34

3D buildings

What this tests: Making 3D buildings with roofs from their outlines, and saving them as a 3D model file (.glb).

What you get: Buildings with the expected volumes, and a .glb that opens in a 3D viewer.

1. Add footprints .geojson. Symbology › **Make 3D buildings**. Tilt the map.
2. Select House A. Then select the Office and press 2F; then set its roof to Hip and roof rise 2.
3. Export as a 3D model (.glb) and open it in any 3D viewer (e.g. the Windows 3D Viewer or macOS Preview).

Expect: House A has a gable roof along its long side: **1,500 m³**, top at **9 m**; the Store is a pyramid (about 849 m³); the Office took its height from 3 floors: 9 m and 1,296 m³. After 2F it is 6 m tall, then a hip roof 2 m higher. The .glb opens with the three buildings.

Pass ☐ Fail ☐ Notes

11. Desktop: select, filter, calculate, join

DT-35

Select by attributes

What this tests: Picking features by their values with a written condition (e.g. pH over 8.5).

What you get: The expected number of features selected.

1. `[ph] > 8.5; then [surveyor] = 'A. Kamara' AND [condition] = 'turbid'; then [site_id] IN ('W01', 'W02', 'W03')`.

Expect: 4, 7 and 3 features; the builder shows the count as you type.

Pass ☐ Fail ☐ Notes

DT-36

Select by location

What this tests: Picking features by where they are, e.g. households within 500 m of a clinic.

What you get: The expected counts.

1. Households within **500 m** of clinics. Then water samples within the study area.

Expect: 13 households; 30 samples.

Pass ☐ Fail ☐ Notes

DT-37

Filter

What this tests: Hiding features that don't match a condition, everywhere at once (map, table, export).

What you get: Only clear samples shown and exported.

1. Filter water_samples to `[condition] = 'clear'`; open the attribute table; export it.

Expect: Only 16 show, in the map, the table and the export; a funnel shows next to the layer.

Pass ☐ Fail ☐ Notes

DT-38

Join and calculate

What this tests: Joining a table to a layer by a shared value (site_id) and filling a new field from a formula.

What you get: 38 of 40 joined, and a filled ph_class field.

1. Join lab_results.csv to water_samples by site_id. Select [nitrate_mg_l] IS NULL.
2. Calculate field **ph_class** = if([ph] > 8.5, 'high', 'ok').

Expect: The join says **38 of 40** match before joining; the IS NULL selection is W07 and W23; ph_class is “high” for 4.

Pass ☐ Fail ☐ Notes

12. Desktop: overlay and feature tools

Words used in this section

Word	What it means
Overlay	Combining two layers by where they overlap (clip, erase, intersect, union).
Spatial join	Copying values from one layer to another by where features are.
Delaunay triangles	Triangles joining points so that no point is inside another triangle's circle.

DT-39

Clip, erase, intersect, union

What this tests: Overlay tools that combine two layers: keep what is inside (clip), remove what is inside (erase), keep what is in both (intersect), or everything cut together (union).

What you get: Layers with the expected numbers of features.

1. Clip water_samples to study_area. Erase water_samples with study_area. Intersect water_samples with districts. Union districts with study_area.

Expect: Clip **30** points; Erase **10**; Intersect 40 points with a district name each; Union makes the pieces inside and outside the study area for each district.

Pass ☐ Fail ☐ Notes

DT-40

Symmetrical difference, update, tabulate intersection

What this tests: More overlay tools, and a table of how much of each class (e.g. land use) falls in each zone.

What you get: The expected areas for each district.

1. Overlay › Symmetrical difference: districts and study_area. Then Update: districts updated by study_area.
2. Add fields.geojson. Overlay › Tabulate intersection: zones districts, classes from fields, by **land_use**.

Expect: Symmetrical difference: the four districts with the study area cut out, about **198 ha** in all (400 ha less the 202 ha study area). Update: **5** areas — what is left of each district, and the study area with its own name. Tabulate: North-West maize **39,600 m²** (3.96 %) and bare soil 8,800 m²; North-East cassava 18,000 m² and pond 3,000 m²; South-West bare soil 17,600 m²; South-East pond 8,000 m² and village 15,400 m² (each ± 0.5 %); the pond is not listed in South-West, which it only touches.

Pass ☐ Fail ☐ Notes

DT-41 Summarize within and spatial join

What this tests: Counting or adding up what falls inside each area, and copying an area's values onto the points inside it.

What you get: The expected counts per district.

1. Summarize within: districts with water_samples, sum of turbidity. Spatial join: water_samples with districts.

Expect: Counts: North-West **11**, North-East **8**, South-West **10**, South-East **11**.

Pass ☐ Fail ☐ Notes

DT-42 Near, identity, split, distance matrix, merge, append

What this tests: Tools that relate features by distance and combine or split layers.

What you get: Distances, 4 split layers, a 60-line matrix and an 8-point merge.

1. Near: households to clinics. Identity: households by districts. Split water_samples by districts. Distance matrix: households to 2 nearest clinics. Merge clinics and candidate_sites. Append boreholes into an editable point layer.

Expect: Near adds a distance to every household; Split makes **4** layers; the matrix has 60 lines; Merge has 8 points with their source.

Pass ☐ Fail ☐ Notes

DT-43 Dissolve, eliminate, Delaunay, bounding geometry

What this tests: Joining areas into one, triangles between points (Delaunay), and the smallest shape around features.

What you get: One 400 ha area, the expected triangles and 1 km squares.

1. Dissolve districts into one. Delaunay on water_samples. Bounding geometry: smallest rectangle around each district; convex hull of all households.

Expect: One area of 400 ha; Delaunay **71** triangles; each district's rectangle is 1,000 × 1,000 m.

Pass ☐ Fail ☐ Notes

DT-44 Shape tools

What this tests: Tools that make new shapes from old ones: points along lines, extra corners, simplifying, random points, crossings and a centre point.

What you get: Each new layer as described.

1. Toolbox: **Points along lines** on roads every 100 m; **Densify** districts every 50 m; **Simplify** the result at 5 m; **Random points in areas**, 10 per district; **Line crossings** on roads; **Mean centre** of water_samples.

Expect: Points every 100 m along each road (no duplicate at the ends); densified districts with a corner at least every 50 m, simplified back close to the original; **40** random points, 10 inside each district; a point at each road crossing; one mean-centre point in the middle of the samples.

Pass ☐ Fail ☐ Notes

DT-45

Address search

What this tests: Finding places by name or address (geocoding) and putting addresses from a table on the map.

What you get: The hospital found, and points for the real addresses.

Data: addresses.csv · **Needs:** Internet.

1. Map › Go to XY: type **Connaught Hospital** and pick it. Then Toolbox › Web › **Geocode addresses** on addresses.csv (field address, country Sierra Leone).

Expect: The hospital is found in central Freetown and the map flies there. The geocoding adds points for the three real places (with what each matched) and lists the made-up one as not found; it takes about a second per address.

Pass ☐ Fail ☐ Notes

13. Desktop: network analysis

Words used in this section

Word	What it means
Network analysis	Working with travel along roads rather than in straight lines.
Cost	What a route tries to keep small: distance or time.
OD matrix	A table of travel from many origins to many destinations.

DT-46

Route by distance and by time with a one-way street

What this tests: Finding the best route along roads, by shortest distance or quickest time, obeying one-way streets.

What you get: Route lengths and times that match the expected values.

1. Network analysis › Route on roads: stops Clinic A then Clinic B, cost length. Then cost time, speed from **speed_kmh**, one-way from **oneway**. Then B to A.

Expect: Length about **2.40 km**; A → B **3.84 min**; B → A **3.52 min** (B → A can use the westbound-only block of Main Road; A → B can't).

Pass ☐ Fail ☐ Notes

DT-47

Rush hours and U-turns

What this tests: Routes that change with traffic at different times of day, and with or without turning round in the road.

What you get: Different routes at 08:00 and 12:00, and with and without U-turns.

1. Route Clinic A → Clinic B by time, speed from **speed_kmh**, one-way from **oneway**; Leave at **08:00**; Add a period 07:00–09:00, speed then from **speed_am**. Then Leave at 12:00.
2. Add **uturn_stops.geojson**. Route through its three stops by distance: U-turns Allowed anywhere, then **Never**.

Expect: At 08:00 about **4.8 min** along the side streets (the main roads crawl at 15 km/h then); at 12:00 back to **3.84 min** by the main roads. U-turns allowed: **150 m** (turning round at stop 2); never: about **1,650 m** (on to the junction and round a block to come back in from the west).

Pass ☐ Fail ☐ Notes

DT-48 Service area, closest facility, OD matrix

What this tests: How far you can travel in a set time (service area), which clinic is nearest for each household, and a table of travel between many places (OD matrix: origins and destinations).

What you get: Travel-time bands, the nearest clinic for each household and a 2×6 table.

1. Service area around the clinics: 2, 4 and 6 minutes. Closest facility: households to clinics, by length. OD matrix: clinics $\times$ candidate_sites.

Expect: Three bands spreading along the roads, wider along the main roads; closest clinic: A for 18 households, B for 12; a 2×6 table.

Pass ☐ Fail ☐ Notes _____

DT-49 Location-allocation

What this tests: Choosing the best place for a new facility so people travel the least.

What you get: One new site chosen, with each household linked to its site.

1. Demand: households weighted by people. Facilities: clinics (existing = yes) and candidate_sites. Least total travel, 1 new site.

Expect: Both clinics are kept; one candidate is chosen and every household is linked to its site.

Pass ☐ Fail ☐ Notes _____

DT-50 Vehicle routing

What this tests: Planning delivery rounds for several vehicles with load limits and time windows.

What you get: Every order served on time, no vehicle overloaded.

1. Depots: depot (vehicles from **vehicles**, capacity from **capacity**), shift 08:00–17:00. Orders: orders, load, service_min, not_before, arrive_by. Search 10 s.

Expect: All 12 orders served (total load $18 \leq 2 \times 10$); no vehicle carries more than 10; orders O01–O04 arrive between 08:00 and 10:00, O05–O08 between 09:00 and 12:00.

Pass ☐ Fail ☐ Notes _____

14. Desktop: toolbox and scripts

DT-51 History, batch and a model

What this tests: Re-running tools from history, running one tool on several layers at once, and chaining tools into a model you can run again.

What you get: The same result from the model as by hand.

1. Toolbox: run Buffer 100 m on clinics; History › Again. Batch: Clip water_samples and households to study_area at once.
2. Model: select [ph] > 8.5 → buffer 500 m → clip households to the buffers. Save it; run it again.

Expect: History lists each run with its inputs; the batch makes two layers; the model gives the same result as doing the steps by hand.

Pass ☐ Fail ☐ Notes

DT-52 Python script

What this tests: Running your own Python script inside Fieldra to automate work.

What you get: The script's printout and two new layers.

1. Script › Python: open script_example.py (rename the layer in it if yours is named differently); Run.

Expect: It prints **40 samples; 4 with pH over 8.5** and their site ids; adds "High pH" and a 100 m buffer layer. The first run takes a few seconds.

Pass ☐ Fail ☐ Notes

DT-53 JavaScript and the computer's Python

What this tests: Scripts in JavaScript, and running Python with the libraries installed on the computer.

What you get: The same output from both.

Needs: Python 3.8+ installed for the second part.

1. Run script_example.js. Then the Python script with Run with › Python on this computer.

Expect: The same output; with the computer's Python, its own libraries (e.g. pandas) can be imported.

Pass ☐ Fail ☐ Notes

What this tests: Plugins add new tools and buttons to Fieldra Desktop, written in JavaScript or Python by you or others. Their tools work like the built-in ones, also in models and scripts.

What you get: A plugin made, its tool and button working, then turned off and removed.

1. Analysis › Geoprocessing › **Plugins** › Manage plugins › **Create a plugin:** name “Test tools”, JavaScript. Close the folder that opens.
2. Add pipes.geojson. Plugins menu: **Count features**. Then **Long lines...**: Lines = pipes, longer than **200** m; Run.
3. Toolbox › Model: Long lines (pipes, 200) → Buffer 20 m of its result; Run. Script (JavaScript):

```
await fieldra.run('test-tools/long-lines', {input: 'pipes', metres: 200}).
```
4. Create a second plugin “Py tools” in Python; run its Long lines on pipes, 200 m. Turn Test tools off, then on. Remove Py tools.

Expect: Test tools shows On with 1 tool and 1 command. Count features shows each layer's count in a message. Long lines makes a layer of **2** of the 4 pipes, with length_m about **251, 300**. The model and the script give the same pipes (the model buffers them). The Python tool gives the same pipes (its first run takes a few seconds). Turned off, the tools leave the toolbox and the Plugins menu; on again, they come back. Remove asks first (black button), then the plugin is gone.

Pass ☐ Fail ☐ Notes

15. Desktop: rasters, TIN and hydrology

Words used in this section

Word	What it means
DSM / DEM / DTM	Elevation images: DSM includes buildings and trees; DTM (or DEM) is the bare ground.
Interpolation / kriging	Making a surface from point measurements; kriging is a statistical method that also gives its uncertainty.
Watershed	The area of land that drains to one point.
TIN	A ground surface made of triangles between measured points.
NDVI	A number from -1 to 1 for how green vegetation is, from red and near-infrared light.

DT-55**Interpolation**

What this tests: Interpolation makes a continuous surface (an image) from measurements at points, e.g. pH everywhere from 40 samples. Kriging is a statistical method that also says how sure it is (standard error).

What you get: A pH surface rising west to east, and an error surface.

1. Interpolate water_samples ph: Kriging, semivariogram Best fit, with standard error; then universal (linear); then IDW.

Expect: pH rises from west (about 6.56) to east (about 8.91); the error surface is larger far from samples; cross-validation shows a small error (about 0.1, well under 0.3).

Pass ☐ Fail ☐ Notes

DT-56**Empirical Bayesian kriging**

What this tests: A more automatic kind of kriging that allows for doubt in its own settings.

What you get: A surface like ordinary kriging with a larger, more honest error.

1. Interpolate water_samples ph: Kind **Empirical Bayesian (EBK)**, 100 points per local model, 100 simulations, with standard error.
2. Again on turbidity_ntu with **Work on the logarithm of the values** ticked.

Expect: One local model (40 points fit in one group); the message gives the spread of the semivariograms and a cross-validation error of about 0.17 for pH; the surface looks like ordinary kriging's, with a larger standard error (about twice, as it includes the doubt about the semivariogram). On turbidity with logs every estimate is above 0. With fewer than 5 points, or a log of a value at or below 0, a clear message instead.

Pass ☐ Fail ☐ Notes

DT-57**Co-kriging**

What this tests: Kriging helped by a second, cheaper measurement taken in more places (here conductivity helps nitrate).

What you get: A more detailed surface with a lower typical error than kriging alone.

1. Add soil_lab.geojson (30 lab nitrate samples) and soil_ec.geojson (300 conductivity readings, 30 of them at the lab sites).
2. Interpolate soil_lab nitrate_mg_kg: Kind **Co-kriging**, second value from soil_ec, ec_ms_m; with standard error. Then the same with ordinary kriging.

Expect: The result card says nitrate and conductivity correlate about **0.99** at the **30** places with both, and the typical error is **lower than with kriging alone** (the card gives both and the percentage). The co-kriged surface shows the fine pattern of the conductivity readings; kriging alone is smoother.

Pass ☐ Fail ☐ Notes

DT-58**Gaussian simulation (risk maps)**

What this tests: Many equally likely versions of the surface, used to map the chance that a value is above a limit.

What you get: Mean, spread, one version and a probability map.

1. Interpolate soil_ec ec_ms_m: Kind **Gaussian simulation**, 50 realisations, threshold 200.

Expect: Four images: the **mean** (close to the kriged surface), the **spread** (small near readings, larger between them), one realisation (patchier, still matching the readings) and the **probability** above 200 (near 0 where readings are low, near 1 where they are high). Running it again gives the same result.

Pass ☐ Fail ☐ Notes

DT-59**Full detail for a large image**

What this tests: That large images can be processed at full detail when asked, not only a faster reduced version.

What you get: A sharper slope image at full detail.

1. Open slope on the elevation model with Detail **standard**, then with **full**.

Expect: Both run; on an image wider than 2,048 cells the full result is sharper (its cell size is the image's own) and takes longer; the choice is kept for next time.

Pass ☐ Fail ☐ Notes

DT-60**Slope, hillshade, calculator, reclassify, viewshed**

What this tests: Basic tools for elevation models (DSM/DEM: images where each cell holds a height): steepness, shaded relief, maths between images, grouping values into classes, and what can be seen from a point.

What you get: Slopes, height differences and classes as expected.

1. Drone › Orthomosaic: add dsm_2026-01.tif and dsm_2026-03.tif. Raster tools: slope and hillshade of 2026-01.
2. Calculator: 2026-03 minus 2026-01. Reclassify the slope: 0–3° flat, 3–15° moderate. Viewshed from a point on the middle ridge.

Expect: Slope about **8.7°** on the flanks and **1.7°** along the valley floors (and steep on the stockpile); the difference is about **–6 m** where pile 1 was and **+4 m** at pile 2, 0 elsewhere; most of the model is “moderate”.

Pass ☐ Fail ☐ Notes

DT-61**Contours, section and volume**

What this tests: Contour lines, a cross-section along a line, and the volume of a stockpile above a base.

What you get: Contours 45–80 m, a W-shaped section and the expected volume.

1. Click dsm_2026-01: Contours every 5 m. Add section_line.geojson and click it › Section.
2. Add stockpile_outline.geojson; click it › Volume, base **best-fit plane**; Save to the area's attributes.

Expect: Contours from 45 to 80 m; the section is a “W”: about **71.5 m** at both ends and the middle, **49.1 m** in the two valleys; the pile above the base is **1,414 m³** (± 3 %).

Pass ☐ Fail ☐ Notes

DT-62**Compare two dates**

What this tests: Comparing elevation models or photos from two dates to see what was removed or added.

What you get: The volumes taken away and added.

1. Drone › Compare dates: Before dsm_2026-01, After dsm_2026-03; Change. Then the orthophotos: Swipe and Side by side.

Expect: Removed about **1,414 m³** (pile 1), added about **419 m³** (pile 2), net about **-995 m³**; red where pile 1 was, blue at pile 2.

Pass ☐ Fail ☐ Notes

DT-63**Hydrology: streams and watersheds**

What this tests: Where water flows: stream lines and watersheds (the area that drains to a point), with flow path length and the time for rain to reach the outlet.

What you get: Two streams and two watersheds of about 18 ha.

1. Analysis › Hydrology on dsm_2026-01: streams where more than **1 ha** drains; watersheds for pour_points.geojson (snap 30 m).
2. Open the attribute tables of the streams, the watersheds and the longest flow paths.

Expect: Two streams, one down each valley, Strahler order 1, Shreve 1, about **18 ha** each at the bottom; two watersheds of about **18 ha** split along the middle ridge. Each watershed: longest flow path about **748 m** (± 10), relief about **40.3 m**, slope **5.4 %**, time of concentration (Kirpich) about **9.8 min**; the longest flow path is drawn in red from the far corner, across the flank and down the valley.

Pass ☐ Fail ☐ Notes

DT-64**Hydrology: basins, flow length, D-infinity and wetness**

What this tests: More drainage results: basins, flow direction and distance, and a wetness index (where the ground stays wet). D-infinity is a smoother way to split flow between cells.

What you get: Two basins and the flow images as described.

1. Hydrology on dsm_2026-01 again: untick streams and watersheds; tick Basins (leave out smaller than 1 ha); Images: **D-infinity**; tick flow accumulation, flow direction, flow length downstream and upstream, and wetness index.
2. Click the images on the east side of the east valley (between the valley and the right edge) and on its west side.

Expect: Two basins of about 18 ha, one per valley. D-infinity direction about **191°** on the east side of a valley and about **349°** on its west side (downhill across the flank and a little south); flow length downstream up to about 748 m at the far corners; upstream largest at the outlets; the wetness index highest along the valley floors; the D-infinity accumulation spreads smoothly over the flanks instead of in D8's straight lines.

Pass ☐ Fail ☐ Notes

DT-65**Hydrology: depressions on a graded platform**

What this tests: Finding puddles (low spots that hold water) on a levelled site, with their area and volume.

What you get: Two depressions with the expected areas and volumes.

1. Add `platform_dsm.tif`. Hydrology: untick everything else; tick Depressions, at least 5 cm deep and 2 m² large.

Expect: Two depressions: about **70 m²** holding **7.9 m³**, 22 cm deep, and about **24 m²** holding **1.3 m³**, 10 cm deep ($\pm 5\%$); the third puddle, only 2 cm deep, is left out; a depth image shows the water in each.

Pass ☐ Fail ☐ Notes

DT-66**Hydrology tools in the toolbox**

What this tests: The same drainage tools used one by one from the toolbox and in a model.

What you get: The same watersheds as before.

1. Toolbox > Hydrology > Watershed: `dsm_2026-01`, `pour_points`. Then a model: Fill → Flow accumulation (D-infinity) → Reclassify (0-1=0; 1-1000=1).

Expect: The same two 18 ha watersheds as the Hydrology window; the model gives an image where 1 marks cells draining more than 1 ha.

Pass ☐ Fail ☐ Notes

DT-67**TIN, edits and rebuild**

What this tests: A TIN (triangulated irregular network) is a ground surface made of triangles between measured points. This test builds one, edits it and rebuilds.

What you get: The expected number of triangles, volume, and change after the edit.

1. Add `spot_heights.csv`. Analysis > Raster > TIN: heights from **z**; triangles, contours every 1 m, an elevation model; volume above **20 m**.
2. In "... TIN edits", add a point on the centre spot S061 (about 8.44963, -13.22467) with **z 33.50**, action **height**. Open TIN with the edits layer selected > Rebuild.
3. Add `kerb_breakline.geojson` as breaklines (ELEV) and draw a hole in "... TIN holes"; Rebuild.

Expect: About **228** triangles (± 3); contours 21–27 m; volume above 20 m **35,288 m³** ($\pm 0.5\%$); after the edit **+829 m³** ($\pm 2\%$); the same layers update in place; the hole leaves a gap and lowers the volume.

Pass ☐ Fail ☐ Notes

DT-68**Image tools**

What this tests: Tools for satellite and drone images: NDVI (a number for how green and healthy vegetation is), averages per field, and classifying the image into land cover.

What you get: NDVI per field matching the expected values.

1. Add `multispectral_BGRN.tif` and `fields.geojson`. NDVI (red band 3, near infrared band 4). Zonal stats of NDVI by fields.
2. Classify: k-means, 5 classes; then from training areas (fields, by `land_use`). Focal mean 3×3 ; resample to 5 m; clip to fields; raster to polygons on the classes.

Expect: Mean NDVI: Maize **0.8**, Cassava **0.69**, Bare soil **0.15**, Pond **-0.43**, Village **0.05** (± 0.03); the classes follow the five fields.

Pass ☐ Fail ☐ Notes _____

DT-69**Cost distance, suitability, change detection, features to raster**

What this tests: Finding the easiest path over the ground, scoring places for suitability, finding change between dates, and turning areas into an image.

What you get: A path that avoids steep ground and change at the two piles only.

1. Cost distance over the slope from Clinic B; least-cost path to a household. Suitability: slope (low better) and NDVI. Change detection on the two DSMs, threshold 0.5 m. Features to raster: districts by population, 10 m.

Expect: The path avoids the steep flanks; change detection shows a decrease at pile 1 and an increase at pile 2 only.

Pass ☐ Fail ☐ Notes _____

DT-70**Georeference a scanned plan**

What this tests: Georeferencing: placing a scanned paper plan on the map by matching marked points to the same places on the map.

What you get: The plan lying on the roads with small errors.

1. Raster > Georeference scanned_plan.png with roads shown. Click cross A on the picture, then the same road junction on the map (A: 695200 E 935200 N); repeat for B, C and D.

Expect: Each point's error is a few metres or less; the GeoTIFF lies on the roads.

Pass ☐ Fail ☐ Notes _____

DT-71**Satellite: Sentinel-2, cloud masks and a cloud-free composite**

What this tests: Downloading free satellite images, hiding clouds and their shadows, and combining several dates into one cloud-free image.

What you get: A scene with clouds masked and a cloud-free composite.

Needs: Internet.

1. Zoom to Freetown. Satellite › Sentinel-2, the last 6 months, cloud at most 60 %. Pick a scene with some cloud; tick Leave out clouds and their shadows; NDVI.
2. Then A cloud-free composite of the 5 clearest, true colour.

Expect: Scenes listed clearest first with previews; the single scene arrives with its clouds blank and says what share was left out; the composite has no cloud (or says what share was cloudy on every date) and is named with its first and last date.

Pass ☐ Fail ☐ Notes

DT-72**Satellite: Landsat history**

What this tests: Older free satellite images (back to the 1980s) to see change over decades.

What you get: Early-1990s images to compare with today.

Needs: Internet.

1. Satellite › Landsat, 1 January 1990 to 31 December 1995, cloud at most 40 %. Add the clearest as true colour; then a composite of 5 as NDVI.

Expect: Landsat 4 or 5 scenes from the early 1990s, with their mission shown; 30 m pixels; compare with today's Sentinel-2 image (swipe in Compare dates) to see how the city has grown.

Pass ☐ Fail ☐ Notes

16. Desktop: statistics, sample design and chainage

Words used in this section

Word	What it means
Hot spot	A place where high values group together more than chance would explain.
Regression / R^2	How one value depends on others; R^2 says how much of the variation is explained ($1 = \text{all}$).
Chainage	Distance along a road or line from its start, e.g. 0+100 is 100 m.

DT-73**Sample design and Thiessen**

What this tests: Planning where to sample (random points, grids, hexagons) and Thiessen polygons: the area closer to each point than to any other.

What you get: Repeatable random points, a grid and two Thiessen areas.

1. Sample design: 30 random points in study_area, at least 50 m apart, seed 42; again with seed 42. Grid of points every 200 m. Hexagons of 10 ha.
2. Thiessen polygons of the clinics, cut to the districts.

Expect: The same 30 points twice; a regular grid; Thiessen gives two areas split along the line halfway between the clinics.

Pass ☐ Fail ☐ Notes

DT-74**Hot spots and clusters**

What this tests: Statistics that find where high or low values group together more than chance would explain.

What you get: Hot spots along Main Road and a significant clustering result.

1. Hot spots on turbidity_ntu (fixed distance, then Optimised). Clusters & outliers on turbidity_ntu.

Expect: Hot spots (red) along Main Road, cold spots (blue) far from it; Moran's I positive and significant ($p < 0.05$); high-high clusters near the road.

Pass ☐ Fail ☐ Notes

DT-75**Regression and GWR**

What this tests: Regression: how much one value (turbidity) depends on others (distance to road, rainfall). GWR lets that relationship vary from place to place.

What you get: Coefficients and R^2 matching the expected values.

1. Regression: turbidity_ntu from dist_road_m and rainfall_mm. Then GWR.

Expect: dist_road_m about **-0.0097** and rainfall_mm about **+0.035**, both $p < 0.05$; R^2 about **0.959**; VIF low (under 2); GWR local coefficients vary only a little.

Pass ☐ Fail ☐ Notes

DT-76**Chainage**

What this tests: Chainage: distance along a road from its start (written like 0+100 for 100 m), and offset: how far to the side of it.

What you get: Points every 100 m and chainages for samples near the road.

1. With the merged Main Road from DT-25: points every 100 m. Then chainage and offset for water_samples near the road; tick Measure from the other end.

Expect: Points at 0+000, 0+100 ... 2+000 (and the end); offsets + on one side, - on the other; chainages reverse with the other end.

Pass ☐ Fail ☐ Notes

What this tests: Placing things on roads by their distance along the road (e.g. a pothole at km 0.5), as road agencies record them.

What you get: Events at the right places and stretches.

1. Analysis › **Routes & events:** route lines roads, route id field name, measures By length, in km.
2. Do: Place point events from `road_events.csv` (route, km, offset `offset_m`). Then line events from `road_surface.csv` (from_km, to_km).
3. Set by posts: `km_posts.geojson` (measure km). Place a point at Main Road km 13.5 (a one-row CSV, or edit `road_events`).
4. Locate features along routes: `water_samples`, within 20 m. Measure ticks every 0.5 km.

Expect: Pothole on Main Road at km 0.5: 8.456443, -13.233263; blocked drain at km 1.25, 4 m right of the road (south, as Main Road runs west to east): 8.456376, -13.226453; broken sign 3 m left of Hospital Road at km 0.2; the row at km 2.4 is listed as beyond the road (it is 2 km long). Gravel from km 0.3 to 1.1 is an 800 m line. With the posts (km 12 at the west end, 14 at the east), km 13.5 is 8.456402, -13.224182. Located samples get `route_id`, measure and `offset_m`; ticks every 0.5 km.

Pass ☐ Fail ☐ Notes

17. Desktop: layouts, dashboard and photos

Words used in this section

Word	What it means
Placement guide	A nonprinting alignment line dragged from a page ruler; it is kept with the layout or template.
Inset / extent indicator	A smaller overview map with an outline showing the area covered by a linked main map frame.
Highlight	A colour behind selected words; removing it does not remove the text colour.

What this tests: Making printed maps with grids, an inset map, tables and charts, and a series with one page per area.

What you get: A PDF map and a 4-page series.

1. Print layout (Cmd/Ctrl+P): UTM grid, inset map, attribute table pages of `water_samples`, chart of samples by condition, georeferenced PDF with PDF layers.
2. Map series: one page per district, named from name.

Expect: A PDF with the map and extras; coordinates read off in a GeoPDF reader; the series has 4 pages.

Pass ☐ Fail ☐ Notes

DT-79 Curved annotation and a PDF layer per map layer

What this tests: Road names that follow the road, and PDFs where each map layer can be switched on and off.

What you get: A PDF with one switchable layer per map layer.

1. roads › Label options: label by name; Convert labels to annotation. Then Print layout with **PDF layers** on, with water_samples, roads and districts shown; Export PDF.

Expect: The road names follow their roads; selecting one shows the stretch it follows. In Acrobat or Avenza the PDF lists one layer each for water_samples, roads, districts and the base map, each switching on and off.

Pass ☐ Fail ☐ Notes

DT-80 Layout designer

What this tests: Designing a drawing sheet with several map frames at different scales.

What you get: A PDF with both frames at their scales.

1. Share › Designer: plan sheet template, A3. Two map frames: 1:5,000 and 1:500, each with a grid and its own scale bar; text with {date} and {scale}. Export PDF at 200 dpi.

Expect: Both frames print at their scales; the text shows the date and scale.

Pass ☐ Fail ☐ Notes

DT-81 Dashboard

What this tests: A summary page of water-quality results against limits, with a PDF report.

What you get: The expected counts above each limit.

Needs: The dashboard reads survey records: send water samples from a phone, or use the phone records from Part A.

1. Share › Dashboard on a project holding water_samples (as phone records via PD-01 or as survey data) and PH-28's sample.

Expect: Of the 40 test samples, pH over 8.5: **4**; turbidity over 5 NTU: **24**; conductivity over 2500: **3**; outside any limit: **28** (plus W-TEST-001 from the phone). PDF report saves.

Pass ☐ Fail ☐ Notes

DT-82 Survey report designer and repeat groups

What this tests: Repeating sections in forms (questions asked once per household member, like KoboToolbox repeat groups), and laying out the survey report yourself: order, chart types, your own text and extra charts.

What you get: A response with several entries, and PDF and Word reports that match the designer's preview.

1. Survey forms › open the form from PD-15 › Edit form: set a section to **Repeat this section**; save, send to the phone, answer once with two entries; sync.
2. **Report designer:** change the title; show one choice question as a pie chart and hide another; move a question to the top; Add text (a heading and a paragraph); Add chart: a bar chart of a choice question, values Total of a number question. Landscape. PDF report and Word report.

Expect: The response shows both entries (Member 1, Member 2); the preview changes as you edit; both reports follow the preview: order, hidden question, pie, your text, the extra chart and landscape pages. Closing and reopening keeps the design.

Pass ☐ Fail ☐ Notes

DT-83**Photos**

What this tests: Managing the project's photos on the computer.

What you get: Each photo action working.

1. Map › Photos: open one, Save a copy, Remove one; Add project photos; on a point, Add photos.

Expect: Each works; removing asks first.

Pass ☐ Fail ☐ Notes _____

DT-84**Layout rulers, guides and flexible placement**

What this tests: Positioning and resizing map-page elements with rulers, saved guides, grid, snapping and zoom.

What you get: A saved layout that is easy to align, without editing guides appearing in the PDF.

1. Open Layout › Open designer (or Share › Designer). Turn on Rulers; drag one guide from the top ruler and one from the left ruler onto the page.
2. Turn on Grid, choose 5 mm spacing and enable Snap. Move a shape near a guide, the page centre and another element. Hold Alt and place it between grid points.
3. Resize from a left edge and a top corner; hold Shift to preserve proportions. Zoom to 150%, scroll, then Fit page. Save a template and reopen it; drag a guide outside the page to remove it.

Expect: • Rulers follow the page origin and zoom; snapping aligns edges and centres, while Alt permits free placement. Opposite resize edges stay anchored; Shift keeps proportions. • Guides are kept in the saved template. Rulers, placement guides and the dot grid are absent from Export PDF.

Pass ☐ Fail ☐ Notes _____

DT-85**Designer follows the current project map**

What this tests: Copying the current map into a layout frame, refreshing previews and reopening a project-specific design.

What you get: The expected map and linked overview in the preview and PDF, with separate working designs for each project.

1. Pan and zoom the main map, rotate it and hide one layer. Open Designer; select the main frame and press Use current map, then Refresh map previews.
2. Add a linked inset and repeat after moving the main map elsewhere. Export PDF and compare with the preview.
3. Open a second project, open Designer and save a different design. Return to the first project and reopen Designer.

Expect: • Use current map copies the current camera and visible layers. Refresh redraws the previews without moving page elements. The linked inset recenters and keeps an overview of the main frame. • Each project reopens its own working design; another project's saved map extent and layer choices do not replace it.

Pass ☐ Fail ☐ Notes _____

DT-86**Linked inset map and picture placement**

What this tests: An overview that outlines the main map, plus a picture fitted or cropped on the page.

What you get: A PDF that matches the inset extent and image placement in the preview.

Needs: A small picture or logo of your own.

1. Select the main map frame; Insert › Inset map. Move the overview, set its scale, and set its Extent indicator to the main frame.
2. Pan, change the scale and rotate the main frame. Change the outline colour with a preset. Add a small PNG or JPEG through Image / logo; resize from two different corners, then try Fit inside and Fill.
3. Export PDF at 200 dpi. Compare the main map, inset outline and picture with the preview. Hide the linked main frame, then undo; remove it, then undo.

Expect: • The inset outline follows the main frame, including rotation, and is clipped to the inset in preview and PDF. The picture fits or crops as chosen. • A hidden main frame is not outlined; removing it clears the link. Undo restores the previous layout. Rotated frames omit GeoPDF coordinates and coordinate grids.

Pass ☐ Fail ☐ Notes _____

DT-87**Colour presets across layouts and reports**

What this tests: The shared colour palette for layout properties, report accents, text and highlights, including compact notes.

What you get: Colours applied on the first click, correctly marked when reopened and kept in exports and saved designs.

1. In Designer, choose preset colours for page colour, a shape fill/border and an inset outline; reopen each picker. Choose one recent colour and None for the shape fill.
2. Survey forms › Report designer: change chart/heading accent through More colours. Select words in the introduction and in an added text block, then apply preset Text colour and Highlight.
3. Repeat text colour and highlight in a compact result note. Reopen Highlight, then No highlight. Try a custom system colour and a complete six-digit #hex value.
4. Export the layout PDF and the report as PDF and Word. Close and reopen both designers.

Expect: • Brand, grid and recent swatches apply on the first click; reopening marks the current colour. None clears a fill. Custom colours update immediately. • Text selection is retained; colour and highlight appear in the preview and exports. No highlight removes only the background. Saved designs retain the changes.

Pass ☐ Fail ☐ Notes _____

18. Drone mapping

Flight photos are not in the test pack (they are large): use a flight of your own, or one of OpenDroneMap's public sample data sets on GitHub (e.g. odm_data_aukerman, about 80 photos).

Words used in this section

Word	What it means
Photogrammetry	Making maps and 3D models from many overlapping photos.
Orthomosaic / orthophoto	One photo of the whole site, corrected so distances are true like a map.
Point cloud	Millions of 3D points describing the surface.
Overlap	How much each photo shares with the next; mapping needs a lot (about 70–80 %).
RMSE	Root mean square error: the usual single figure for typical error in an accuracy report.

DR-01 Flight check and quick map

What this tests: Checking drone photos before processing: GSD, overlap between photos (they must overlap a lot to make a map), and gaps. The quick map lays the photos out in minutes.

What you get: A check of the flight, and a rough map.

Needs: Drone photos.

1. Drone › Flight photos: choose all JPEGs. Then Drone › Quick map on the same folder.

Expect: GSD, front and side overlap, photos per patch and RTK share; red patches are gaps. The quick map lays the photos in place within a couple of minutes.

Pass ☐ Fail ☐ Notes

DR-02 Photogrammetry

What this tests: Photogrammetry turns many overlapping photos into maps and 3D: an orthomosaic (one map-accurate photo), a DSM (heights) and a point cloud.

What you get: The orthomosaic, DSM, point cloud and a quality report.

Needs: Drone photos; 16 GB of memory; on a Mac, Docker Desktop.

1. Drone › Photogrammetry: This computer › Install the engine (once). Photos; **Fast map**; Process. Then **2D maps** at 5 cm.
2. Quality report; save the PDF. Process once more with **3D model**, then 3D model on the map. Send to phones.

Expect: The orthomosaic, DSM, point cloud and camera positions are added by themselves; the report lists photos used, GSD and reprojection error.

Pass ☐ Fail ☐ Notes

DR-03 Ground control points in photos

What this tests: Marking the GCP targets in the photos so the map is tied to measured positions. A check point is left out on purpose to test the accuracy later.

What you get: Targets marked in 3 photos each; one kept as a check point.

Needs: Drone photos with surveyed targets.

1. Add ground control points › Load points (a GCP file for the photos). Mark each in 3 photos (N for the next). Make one a check point.

Expect: Dots turn green at 3 marks; marks are kept if you close and reopen; the check point is left out of processing.

Pass ☐ Fail ☐ Notes

DR-04 Orthophotos, bands and tiles

What this tests: Showing orthophotos and multispectral images (which also record light the eye can't see, such as near infrared), with band combinations.

What you get: A sharp orthophoto and a colour-infrared view.

1. Add ortho_2026-01.tif; change opacity. Add multispectral_BGRN.tif › Display: bands 4-3-2 (colour infrared), stretch 2–98 %; Reset.

Expect: Sharp at every zoom once tiled; in colour infrared the maize and cassava are bright red and the pond dark.

Pass ☐ Fail ☐ Notes

DR-05 Recover a partial result import

What this tests: A result import interrupted by a controlled storage failure.

What you get: Visible progress, a useful error, a retry and results in the correct project.

Needs: A small flight, processing engine and controlled failure setup. Record this as not run if the setup is unavailable.

1. In a disposable test project, complete a small photogrammetry job. Watch the current file and saved-file count as results are added.
2. Using a controlled failing-storage test setup, make one output import fail. Restore storage, then click **Retry adding to map**. Do not fill a production disk to run this test.
3. Switch to another project during the import; return to the original project. Save the quality report and a 3D output after the map layers are added.

Expect: The error remains visible until retry. Previously recorded files are skipped, the retry finishes, and imported layers remain in the original project. Original downloads remain available.

Pass ☐ Fail ☐ Notes

DR-06 Interrupted jobs and 3D deliverables

What this tests: Cancellation, restart and complete 3D export archives.

What you get: Terminal canceled jobs, explicit interrupted-upload errors and usable 3D outputs.

Needs: A small flight and processing engine; record actual outputs and errors.

1. With a disposable job, cancel during upload or download; reopen Fieldra. Also close during a separate upload and reopen.
2. Run a job that produces only a 3D deliverable. Save and unpack its 3D Tiles archive; open the tileset in a compatible viewer.
3. Save any originals needed, then remove the disposable job after imports finish.

Expect: Canceled jobs stay canceled. Interrupted uploads show a failure rather than polling forever. 3D-only jobs finish; referenced GLB payloads are present. Removal clears retained job downloads.

Pass ☐ Fail ☐ Notes

DR-07 Ground control input validation

What this tests: Invalid point files, coordinate choices and folder changes.

What you get: Clear rejection of missing heights or duplicate names, with marks kept in their own folder.

1. Try a point file with a missing height, then one with duplicate point names. Correct each file and choose its actual CRS.
2. For projected coordinates near the origin, explicitly choose the projected CRS. Switch photo folders while saved marks are loading.

Expect: Invalid heights and duplicate names are rejected clearly. Projected coordinates stay in the chosen system. The new folder never receives the previous folder's marks.

Pass ☐ Fail ☐ Notes

DR-08 Large raster and boundary handling

What this tests: Memory use, detailed polygon statistics and temporary-file cleanup.

What you get: Results without the former full-image contrast copy or polygon argument-limit error.

Needs: A representative raster and boundary. Record hardware, elapsed time and memory; there is no universal speed threshold.

1. Use a 16-bit GeoTIFF without overviews in a disposable project. Prepare phone tiles while observing memory, progress and responsiveness.
2. Compare a normal polygon and an equivalent densely sampled boundary (over 250,000 vertices) in full-resolution zonal statistics.
3. Cancel tile preparation through an available task cancellation path, or run the native cancellation regression; inspect temporary and completed outputs.

Expect: Contrast preparation does not allocate a full-image float working copy. Detailed boundaries finish without an argument-limit error. Canceled preparation removes its partial file and preserves previous completed output.

Pass ☐ Fail ☐ Notes

DR-09 Point cloud: open and view

What this tests: Opening a point cloud (millions of 3D points, from LiDAR or photogrammetry) and colouring it in different ways.

What you get: The cloud with the right number of points in the right place.

1. Drone › Point cloud: `site_lidar.las`. Colour by height, class, intensity, real colour. Click a roof point. Then `site_lidar_no_crs.las`.

Expect: 123,440 points over a 100 m square in Freetown; UTM 28N is read from the file; the roof of the big building is about 8 m above the ground; for the copy without a system, Fieldra asks and suggests **WGS 84 / UTM zone 28N**.

Pass ☐ Fail ☐ Notes

DR-10 Ground, noise, buildings, trees and wires

What this tests: Sorting the points into classes: ground, noise (wrong points), buildings, vegetation and wires.

What you get: Each class found as expected.

1. LiDAR tools: Classify ground (largest building 25 m; Find noise first on). Then **Find buildings, trees and wires** (smallest building 20 m²). Colour by class.

Expect: 40 low noise points (class 7); ground everywhere else on the ground; 2 buildings (class 6) with roofs and walls; the power line across the north (class 14, about 400 points, 10 m up); trees high vegetation, the bushes in the south-east low or medium.

Pass ☐ Fail ☐ Notes

DR-11 Building footprints

What this tests: Drawing building outlines and heights from the classified point cloud.

What you get: Two outlines with the expected areas and heights.

1. Building footprints.

Expect: A layer with 2 outlines: about 300 m² and 8.0 m high, and about 120 m² and 6.0 m high; drawn as blocks in 3D.

Pass ☐ Fail ☐ Notes

DR-12 Learn from examples

What this tests: Teaching the classifier by marking a few examples, then letting it class the rest.

What you get: Over 90 % right on points it had not seen.

1. Set class in the selected area: draw small areas on a roof (building), a tree (high vegetation), the open ground (ground) and the line (wire). Learn from my examples.

Expect: It reports how often it was right on points it had not seen (over 90 %); the rest of the cloud is classed to match.

Pass ☐ Fail ☐ Notes

DR-13 DTM, DSM and saving classes

What this tests: A DTM is the bare ground; a DSM includes buildings and trees. This test makes both and saves the classes.

What you get: A DTM without the buildings and a DSM with them.

1. Make DTM 0.5 m and Make DSM 0.5 m. Save classes. Export LAS...; open the exported file again.

Expect: The DTM is a smooth slope without the buildings; the DSM has them; the saved classes are there when the file is opened again.

Pass ☐ Fail ☐ Notes

DR-14 A big point cloud

What this tests: That very large point clouds open quickly and fill in detail as you zoom.

What you get: The cloud shown at once, more detail where you zoom.

Needs: A large cloud of your own.

1. Add any LAS or LAZ with more than 3 million points (e.g. from your drone software). Zoom in on one corner; change Points drawn at most.

Expect: It shows at once, a message shows how far the full-detail preparation is; zooming in fills in the detail where you look.

Pass ☐ Fail ☐ Notes

DR-15 Accuracy report

What this tests: How accurate the finished map is: checkpoints measured on the ground are compared with the same points on the map. RMSE (root mean square error) is the usual single figure for the typical error.

What you get: RMSE values, the bad checkpoint flagged, and pass or fail against the targets.

1. Drone > Accuracy report: surveyed checkpoints_surveyed.csv, model checkpoints_model.csv; targets 5 cm horizontal, 10 cm vertical. Save the PDF.
2. Then delete CP9 from the model file and run it again.

Expect: 12 matched; RMSE horizontal **3.2 cm**, vertical **11.9 cm**; **CP9** flagged (40 cm out); vertical target failed; a warning that 12 is fewer than 30. Without CP9 vertical RMSE is **3.0 cm** and passes.

Pass ☐ Fail ☐ Notes

DR-16 Accuracy on the phone

What this tests: The same accuracy check on the phone.

What you get: The same figures as on the computer.

1. Terrestrial Survey > Accuracy with the GCPs imported in PH-08 and a model CSV.

Expect: The same figures as the desktop for the same points.

Pass ☐ Fail ☐ Notes

19. Sign-off sheet

Copy the result of each test here. Anything failed: note the device, the app version and what you saw, with a screenshot.

Test	What	Pass	Fail	Notes
PH-01	Welcome survey and first project			
PH-02	Projects: switch, rename, delete			
PH-03	Separate mobile sample project			
PH-04	Tablet sheets and larger text			
PH-05	Position and accuracy colours			
PH-06	Saving and photos			
PH-07	AI download and suggestion			
PH-08	Import GCPs from a D-RTK 3 export			
PH-09	Planned points and stake out			
PH-10	Receiver and GCP observation (simulated)			
PH-11	Exports for office software			
PH-12	Flight calculator			
PH-13	Flight log and wind warning			
PH-14	Mission area KML for DJI Pilot 2			
PH-15	Laser scanning station log			
PH-16	PPK job			
PH-17	Environmental Survey: every category			
PH-18	Biodiversity: sightings and scores			
PH-19	Biodiversity: point count, transect, quadrat			
PH-20	Land & Scene Analysis			
PH-21	Survey plan: study area and visits			
PH-22	Way Back			
PH-23	Socio-economic: a form with skip logic			
PH-24	Photo Gallery			
PH-25	Compass and Qibla			
PH-26	Road Survey			
PH-27	Land Survey with compass and tape (practical)			
PH-28	Water Survey and label scan			
PH-29	Bluetooth water meter			
PH-30	GNSS averaging, distance and height			
PH-31	Area walk, barometer, sound, magnetic field			
PH-32	Label scanner			
PH-33	Coordinate converter			
PH-34	COGO calculator			

Test	What	Pass	Fail	Notes
PH-35	Sampling points			
PH-36	COGO: bearings and distances together			
PH-37	Sample size for a target precision			
PH-38	Map navigation and layers			
PH-39	Terrain, measure and buffer			
PH-40	Snap and drawing			
PH-41	Layout PDF and image			
PH-42	Offline map			
PH-43	Records and export			
DT-01	Install and first start			
DT-02	Projects and keyboard shortcuts			
DT-03	Command search, sample project, jobs, autosave and history			
DT-04	Project package			
PD-01	Connect a phone and receive			
PD-02	Survey data on the desktop			
PD-03	Nearby Devices: desktop to phone with layers and an orthophoto			
PD-04	Interrupted transfer and deletions			
PD-05	Phone to phone			
PD-06	Edit desktop layers on the phone and sync back			
PD-07	Reshape a desktop area on the phone			
PD-08	Web layers and downloaded features on the phone			
PD-09	Datum grid file sent to the phone			
PD-10	Save phone edits back to a feature service			
PD-11	Team server			
PD-12	Open files, share a layer, record a track			
PD-13	Split an area and overlays on the phone			
PD-14	Land survey on the desktop			
PD-15	Forms from the desktop to the phone			
DT-05	Add the vector files			
DT-06	Tables without coordinates			
DT-07	CAD drawing			
DT-08	Export for ArcGIS and round trip			
DT-09	File geodatabase: domains, subtypes and spatial index			
DT-10	National grids			
DT-11	Site grid			
DT-12	Geoid and heights			

Test	What	Pass	Fail	Notes
DT-13	Web map layers			
DT-14	Feature services (WFS, OGC API)			
DT-15	ArcGIS Online and Enterprise layers			
DT-16	Related tables and attachments			
DT-17	Relationships in a file geodatabase			
DT-18	netCDF variables and curved .gdb edges			
DT-19	Sign-ins in the keychain and edit conflicts			
DT-20	PostGIS: export, edit live, history and review			
DT-21	SQL Server: export, add and edit live			
DT-22	Go to XY, bookmarks and time			
DT-23	Draw, snap, trace, undo			
DT-24	COGO from a deed			
DT-25	Edit tools			
DT-26	Style by attribute and rules			
DT-27	Labels, label classes and annotation			
DT-28	Field rules, subtypes and Check data			
DT-29	Calculated fields			
DT-30	Topology: find and fix			
DT-31	Topology between layers			
DT-32	Drawing options			
DT-33	3D			
DT-34	3D buildings			
DT-35	Select by attributes			
DT-36	Select by location			
DT-37	Filter			
DT-38	Join and calculate			
DT-39	Clip, erase, intersect, union			
DT-40	Symmetrical difference, update, tabulate intersection			
DT-41	Summarize within and spatial join			
DT-42	Near, identity, split, distance matrix, merge, append			
DT-43	Dissolve, eliminate, Delaunay, bounding geometry			
DT-44	Shape tools			
DT-45	Address search			
DT-46	Route by distance and by time with a one-way street			
DT-47	Rush hours and U-turns			
DT-48	Service area, closest facility, OD matrix			
DT-49	Location-allocation			
DT-50	Vehicle routing			

Test	What	Pass	Fail	Notes
DT-51	History, batch and a model			
DT-52	Python script			
DT-53	JavaScript and the computer's Python			
DT-54	Plugins: create, use, turn off, remove			
DT-55	Interpolation			
DT-56	Empirical Bayesian kriging			
DT-57	Co-kriging			
DT-58	Gaussian simulation (risk maps)			
DT-59	Full detail for a large image			
DT-60	Slope, hillshade, calculator, reclassify, viewshed			
DT-61	Contours, section and volume			
DT-62	Compare two dates			
DT-63	Hydrology: streams and watersheds			
DT-64	Hydrology: basins, flow length, D-infinity and wetness			
DT-65	Hydrology: depressions on a graded platform			
DT-66	Hydrology tools in the toolbox			
DT-67	TIN, edits and rebuild			
DT-68	Image tools			
DT-69	Cost distance, suitability, change detection, features to raster			
DT-70	Georeference a scanned plan			
DT-71	Satellite: Sentinel-2, cloud masks and a cloud-free composite			
DT-72	Satellite: Landsat history			
DT-73	Sample design and Thiessen			
DT-74	Hot spots and clusters			
DT-75	Regression and GWR			
DT-76	Chainage			
DT-77	Routes and events			
DT-78	Print layout			
DT-79	Curved annotation and a PDF layer per map layer			
DT-80	Layout designer			
DT-81	Dashboard			
DT-82	Survey report designer and repeat groups			
DT-83	Photos			
DT-84	Layout rulers, guides and flexible placement			
DT-85	Designer follows the current project map			
DT-86	Linked inset map and picture placement			
DT-87	Colour presets across layouts and reports			

Test	What	Pass	Fail	Notes
DR-01	Flight check and quick map			
DR-02	Photogrammetry			
DR-03	Ground control points in photos			
DR-04	Orthophotos, bands and tiles			
DR-05	Recover a partial result import			
DR-06	Interrupted jobs and 3D deliverables			
DR-07	Ground control input validation			
DR-08	Large raster and boundary handling			
DR-09	Point cloud: open and view			
DR-10	Ground, noise, buildings, trees and wires			
DR-11	Building footprints			
DR-12	Learn from examples			
DR-13	DTM, DSM and saving classes			
DR-14	A big point cloud			
DR-15	Accuracy report			
DR-16	Accuracy on the phone			

Tested by	Phone (model, Android/iOS version)	Computer (Mac/Windows version)	Fieldra versions	Date