

Problem 3.3.2

Public Health Emergency Apps

DATE	PORTAL DAY	LESSON	TURN IN
Monday 2026-12-11	Design justification CER	Lesson 3.3 Information Sharing	CER

Where this sits. Day 2 of 2, and the last day of Unit 3. Design, test, revise, and defend.

01 Why today matters

Today the design brief becomes actual screens, and then real people try to use them in front of you.

You finish by defending your design against your own problem statement, which is the only test that matters.

02 What you already know

- You have a design brief, a chosen solution, and a user persona, all checked on Friday.
- You know how to label a sketch with the events each element triggers.
- You know that watching someone hesitate tells you more than asking them if they liked it.

WHAT THIS SHEET WILL NOT TELL YOU

- This sheet does not tell you which screens to build or how many.
- It does not tell you whether your design is good. Your users tell you that, and the rubric tells you whether you documented it.

03 Vocabulary

TERM	WHAT IT MEANS, PLAINLY
flowchart	A diagram of how a user moves between screens and what triggers each move.
universal design	Designing so people of different ages and abilities can all use it, without a separate version for anybody.
accessibility	Whether a person with a disability can actually use the thing. Text size, contrast, labels a screen reader can read, and a non-visual route to any information given only by colour.
prototype	A test version built to try an idea out, not the finished product.
usability testing	Watching real users try to do a real task with your design, and recording where they get stuck.
data security	Keeping stored information from being read, changed, or stolen by people who should not have it.
confidentiality	The duty to keep someone's private information private.

Every term on this list is flagged for the illustrated glossary, scoped to this activity only. Terms are not shared forward or backward between activities, because a definition from a later activity can hand you an answer you were supposed to derive.

04 How to do the work

- 1 Show your checked design brief. No brief, no build.
- 2 Review the three lenses before you draw: user-centered design, universal design, and safety. Write one sentence per lens saying what it means for YOUR app specifically.
- 3 For universal design, go concrete. Is the text big enough? Is anything communicated by colour alone, and if so what is the second route to it? Can someone using a screen reader get the critical information? Can it be used one-handed?
- 4 For safety, answer two questions in writing. Could a user be hurt emotionally by anything your app shows or sends? Could their data be exposed or misused? If your app touches health information, say exactly who may see what.
- 5 Describe your app's major features from the user's point of view. Stay on one goal. An app that tries to do four things does none of them well and you cannot design it in a day.
- 6 List every UI component you will include. Then, for each, answer three questions: what does the user do with it, what data goes in, what data comes back out.
- 7 Sketch the major screens, by hand or digitally. Three or four screens. If you have more than four, your app is doing too much; simplify the purpose, do not shrink the sketches.
- 8 Build the flowchart. Show what each element triggers on its own screen AND how the screens connect to each other. Most teams do the first part and forget the second. Someone has to be able to get back.
- 9 Review your own design against the brief. Point to where each design requirement is met. Point to the critical feature. Then walk through as each type of user you named and check that each one can reach what they need.
- 10 Iterate. Change what the review exposed, and write down what you changed and why.
- 11 Produce a clean final version with a digital drawing tool. Decide colour and font fast and move on; you are graded on function and reasoning.
- 12 Build your presentation. An intro screen with the app name and your team, then a screenshot of each screen. Do not put your talking points on the slides. You explain the purpose out loud; the screens show the design.
- 13 Present to another team or the class. Do not narrate your sketches. Give them a task and watch them try to do it.
- 14 Document the feedback as it happens, including every hesitation and every wrong tap. A wrong tap is the most useful thing you will collect all day.
- 15 Revise based on what you saw and record what changed.
- 16 Close out Unit 3 in your notebook. Answer: what did comparing designs teach you about your own? Why is communication critical in outbreaks and emergencies, and what are the real drawbacks of an app as the main channel? Name three to five skills you built in this unit, with one piece of evidence for each. And name any career from this unit you might actually want, and why.

05 Safety

READ EVERY LINE BEFORE YOU START

- Screen and paper day. Nothing hands-on.
- Invented data only, in every screenshot and every persona. No real names, no real health information, no photographs of real people.
- If your presentation includes a scenario, keep it fictional. Do not build a demo around something that actually happened to someone in this building.

06 Where students get stuck

- Presenting features instead of the problem. Start with who has the problem and what it costs them. Then show how a screen solves it. A feature tour with no problem behind it does not score.
- Narrating your own design during testing. You will fill every gap with your voice and learn nothing. Hand it over, give a task, stay quiet, watch.
- Recording only the praise. 'They liked it' is not feedback. Where did they pause? What did they tap that did nothing? What did they ask you? Those are the findings.
- A flowchart that only covers one screen. The connections between screens are the part users actually get lost in. Show the way back from every screen.
- Leaving accessibility as a sentence you say rather than a thing you designed. Point at the element on the screen that carries it.
- Adding features on Monday. Every new feature is a new screen, a new event, and a new way to get lost. Cut instead.
- Skipping the privacy answer because your app 'does not really use data'. If a user types anything into it, it uses data. Say who can see it.

07 What you turn in

CER

CER, one per team, plus the design package. Claim: your app design solves the problem in your problem statement. Evidence: the specific screens and features that do it, plus what you observed during user testing, including the places people hesitated. Reasoning: why those design choices meet the needs of the user you named, and what you changed after testing and why. The package attached to it is your design brief, your labelled screen sketches, your flowchart, and your presentation. Everything checked against the Problem 3.3.2 rubric. Your Unit 3 reflection goes in your notebook.

08 Check yourself

- ☐ Give two concrete universal design changes you could make to almost any screen. (Vocabulary and step 3)
- ☐ During testing, why is a wrong tap more useful than the words 'that was easy'? (Where students get stuck)
- ☐ Your app lets users report symptoms. Name one privacy decision you have to make and state your answer. (Step 4 and Friday's PII vocabulary)