The right experiment structure

Key idea

- **Measured task = one controlled article per UI** (so you can compute reading rate + time-to-click cleanly).
- **Optional free browsing = allowed, but NOT part of the primary DV** (or analyzed separately).

Recommended flow (within-subject, counterbalanced AB/BA)

1. **Participant info + consent**
2. **Tutorial (20–30s)**: "You will do two short tasks, one per interface. Please do them normally."
3. **UI 1 (A or B)**
    - Start screen: "Task 1"
    - **Scenario card**: "Find and read the Sports article. Then press 'I have finished'."
    - This forces a comparable "habere tıklama süresi" measurement.
    - Participant lands on feed → selects article → reads → finishes → answers post-task questions.
4. **UI 2 (B or A)**
    - Start screen: "Task 2"
    - Same structure, but **use a different topic/article** to reduce memory effects.
5. **(Optional) Free browsing stage (2 minutes)**
    - "Now browse any news freely. You can open multiple articles."
    - After: short preference questions (which UI felt better for exploration?)
6. **Final survey + debrief**

This lets you satisfy:

- instructor's click-time and expectation metrics (controlled scenario),
- your "reading rate" goal (measured on the read article),
- and your intuition that users should be able to read other news (free browsing stage).

How to meet the instructor's specific metrics

He asked for:

1) "A tasarımında habere tıklama süresi / B tasarımında habere tıklama süresi"

That's **time from feed shown → first article click** (or time to click the target article).

Define it explicitly

- `timeToFirstArticleClickSec = ts(first_open_article) - ts(feed_loaded)`

- If you enforce a target:
  `timeToTargetArticleClickSec = ts(open_target_article) - ts(feed_loaded)`

2) “A tasarımında habere girme beklentisi / B tasarımında habere girme beklentisi”

This is basically **perceived willingness/intent** to open an article.

Add 1–2 Likert items right after they view the feed (before clicking), or right after they complete:

- “Bu arayüzde bir habere tıklamak istedim.” (1–7)
- “Bu arayüz beni habere girmeye teşvik etti.” (1–7)

If you want it *pure*, ask it **before they click** (right after the feed loads) with a short popup. But that popup can interrupt behavior, so I’d do it **after** the task and phrase it as a reflection.

Do you need to allow reading other news during the measured task?

For clean analysis: **No, not required.**
But you can allow it in a controlled way:

Best compromise

- During Task 1/2, allow browsing, but your system marks:
  - the **first opened article**
  - the **target article**
  - the **“measured article”** (the one they spent the most time on OR the target)
- Your primary DV still uses the measured/target article only.

Then Free Browse stage is where “read multiple news” is the point.

What to request from your developer (exact checklist)

Ask for **event logging + a simple task controller**.

A) Task controller (so your experiment is structured)

- `conditionOrder`: AB or BA (random assignment)
- `taskId`: Task1 / Task2 / FreeBrowse
- `targetArticleId` per task (e.g., Sports in A, Tech in B)
- A “Start Task” screen that fires `task_started`

- A “Finish Reading” button that fires `task_finished`

B) Required events + timestamps

You need these events (minimal but enough):

Feed

- `feed_loaded` (when the feed is visible)

Article entry

- `article_card_click` (from feed)
  - fields: `articleId`, `articleTopic`, `rankPosition` (where it was on feed)
- `article_opened` (article page loaded)
- `article_closed` (optional)

Reading / progress

- `scroll_depth_update` OR just store:
  - `maxScrollDepthPct`
  - `timeTo50PctSec` (optional)
  - `timeTo90PctSec` (optional)
- `reading_started` (first scroll or 2s after open)
- `reading_finished_clicked` (when they click “I have finished”)

Focus mode (UI-B)

- `focus_mode_toggled_on/off`
- derived metrics:
  - `focusModeUsed`
  - `focusModeTimeSec`

Distractions

- `distraction_click` (clicking side rail, topic chips, ads, etc.)
  - field: `distractionType`

C) Computed metrics to store per task (so analysis is easy)

Store these per task/session:

- `timeToFirstArticleClickSec`
- `timeToTargetArticleClickSec` (if target exists)
- `readingTimeSec` (article_opened → finished_clicked, minus idle if you want)
- `maxScrollDepthPct` (clamped 0–100)
- `completed` (true if scrollDepth≥95 and finished_clicked)
- `distractionClicks`
- `focusModeUsed`, `focusModeTimeSec`

D) Survey timing + questions (exact)

After each UI task, ask:

- Perceived focus (1–7)
- Perceived readability (1–7)
- Expectation / willingness to click (1–7) ☑ instructor metric
- Comprehension MCQ (same difficulty across articles)

After Free Browse:

- Preference: “Which UI made you open more news?”
- “Which UI would you use daily?”

What feature is “the reading rate increasing feature”?

Make it explicit in your report + demo:

- **UI-B Focus Mode** + **reading progress bar** + **reduced distractions** are the feature bundle.
- Your mechanism hypothesis:
    - Focus Mode reduces distractions → increases sustained reading time and completion.

But to satisfy the instructor’s “click-time” request, your UI-B should also have one “entry encouragement” element:

- example: a **prominent Top Story card** with better hierarchy, fewer competing options (Hick), stronger CTA.
  That affects `timeToArticleClickSec`.