MITRE / FLUX NOTES SELECTED CONCEPTS v13.3: LOWERING BURDEN BRAINSTORMING

EDWIN CHOI, DANIEL REEVES, JUHAN SONIN, 6.APR.2018



The following are low-fidelity mockups for reducing clinician burden while using Flux Notes. These are based on the brainstorming design session held on the MITRE campus on 2 April, 2018. These should be seen as <u>rough exploratory concepts</u>. Next week, the concepts with high potential value for reducing clinician burden will be refined into more detailed UI mockups for further clinician review.

The following is an aggregation of the concepts and ideas discussed from the brainstorming design session.

Comment that has been explored within the following low-fidelity mockups.

#### High Level Concepts

- Functionalities of Tech world tools
  - IDEs
  - type checking, autocomplete, views, refactoring, multi-use windows (Microsoft Visual Studios, etc)
  - exploration warranted
- Slack
- Git
- Risk-assessments & eCQMs
  - inserting some clinical guidance and measures
  - prompts to capture the relevant data to perform the right assessments
  - capturing eCQMs
- Multiple entry methods
- narrative, POS, blocks, templates
- Highly customizable
  - extendable
  - open source
  - importable
  - · views etc. templates by physician
  - "empower physicians to do things the way they want"

#### Graphical Assist

- Importing multiple items into note.
  - Some way to select multiple items in the report before pulling into the note
  - maybe a mode selection whereby clicking on structured data elements or sections highlights them, or maybe you tag each via the dropdown and then pull them all in
  - this allows them to be formatted in a smart way, all at once.
- Drag & Drop
  - magnetic poetry
  - connects related ideas to fill in plain text. This is a NLP problem with training on clinical context.
  - Scratch
- POS data entry

#### Narrative Entry

- Single hashtag for a whole topic
- ex: diseaseStatus, then NLP recognizes the following tags without hashtags, perhaps using synonym sets.
- Skip intermediate hashtags and use SHR structure to infer them
- ex: cancer is stable based on CT --> diseasestatus stable
- ambiguity can be eliminated through machine learning
- Tag synonym sets or NLP recognition for tags
- suggestion ranking for dropdown and right context menu
- by context
- order and limit
- · clarity of the shortcut, what is it?
- captures
- pulls from record
- macros run on the record
- external functions (medication orders, etc)
- templates
- Snippets and templates
- customizable summary and HPI etc.
- mad-libs
- · from simple to advanced
- access/show these are customizable
- autocomplete
- skip at and hash alltogether
- whole chunk processing

Dan: this doesn't address pulling data into the note, and also misses the opportunities of seeing capture and pulling as two sides to the same coin

 process each sentence or paragraph, then present and confirm structured data captures

#### Authentication

- Guided Help
- reminders to finish data
- pop up in right tray?
- reminders of next things you may want to do (Turbotax)
- Data bounties
- increase visibility of missing data, stress highest impact
- generate reports on missing data
- maybe clinicians keep running scores -> leaderboards
- Templates to combine all missing data fields into note
- POS for missing data
- "passive" tally of missing data in record, POS entry when desired
- prompted at end
- smart inclusion into note? Or dumped at end?

#### Pre-Encounter

- Customized views
- (overview)
- e.g. templates for data record
- Especially summary section
- replaces/augments the pre-encounter / encounter / post-encounter split?

#### Other

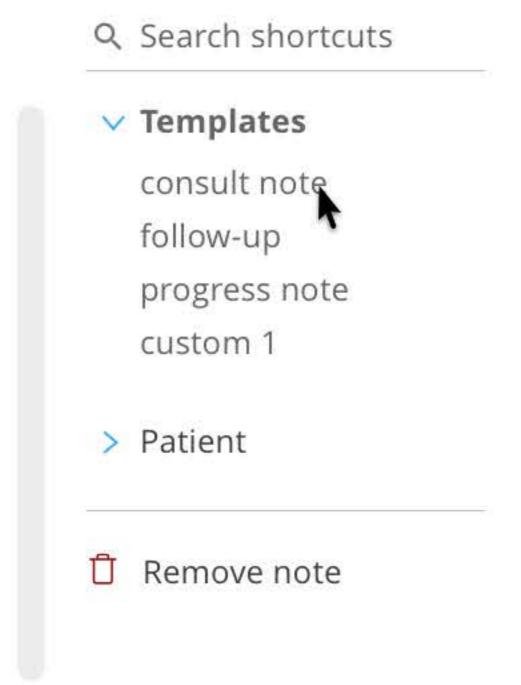
#### Duplicate Entry and Follow Up

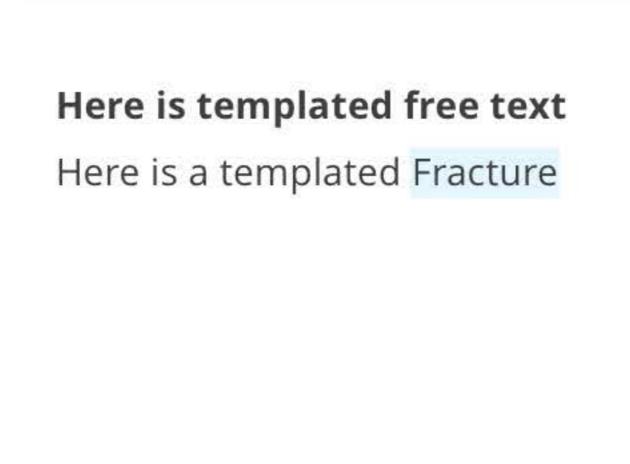
- Related task windows
- When entering data that is relevant to a particular task (ordering medication, ordering lab tests, billing which is always relevant) a helper window is available
- As data is entered, the helper window populates live. The window can be expanded to confirm the relevant data and complete the transaction.
- Maybe upon signing the note, the user is quickly prompted for each helper window to complete or discard the action.

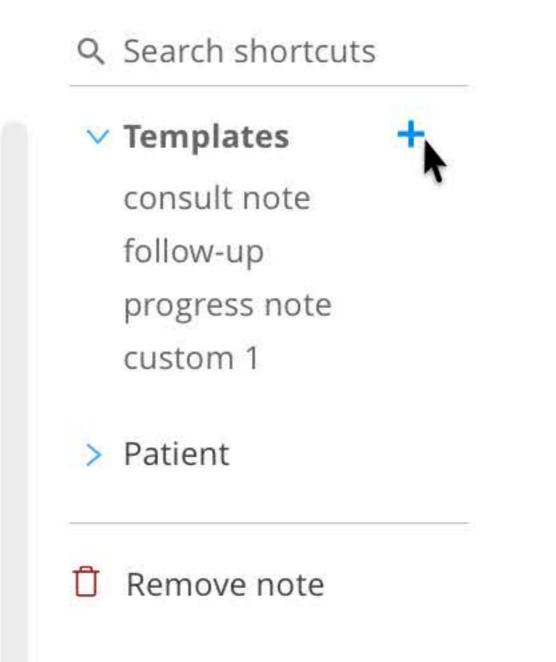
#### Reports

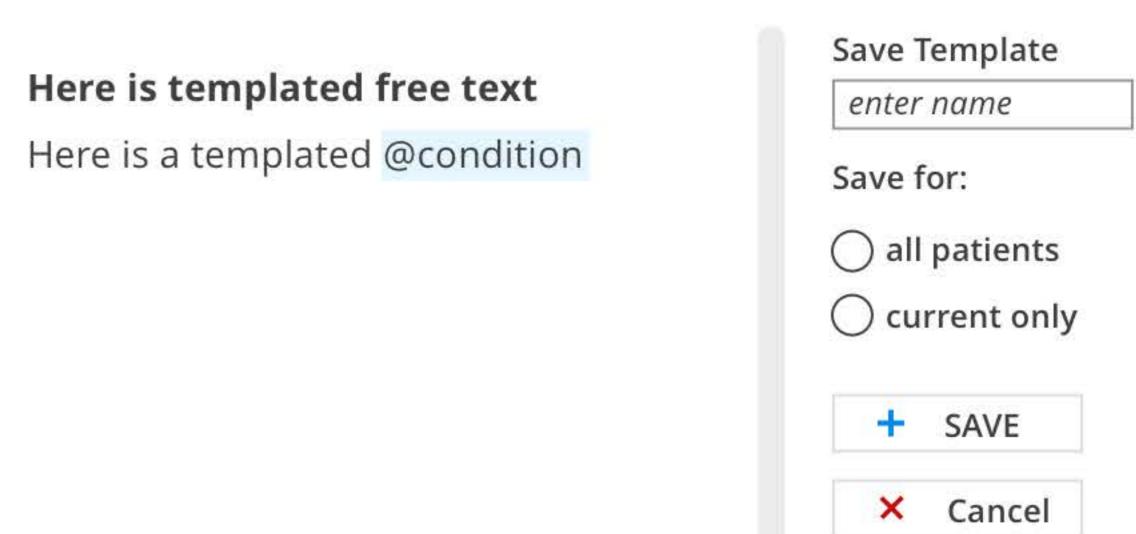
- outpatient reports
  - need more info on this?
- customized or standardized reports, with reporting back to user what is missing
- patient reports
- shortcuts that are action based
- order medication, specialist referral
- right context tray shortcuts predict what the provider needs. Not showing the entire list of macros

# TEMPLATES: New









# TEMPLATES: editing

### Reason for consultation

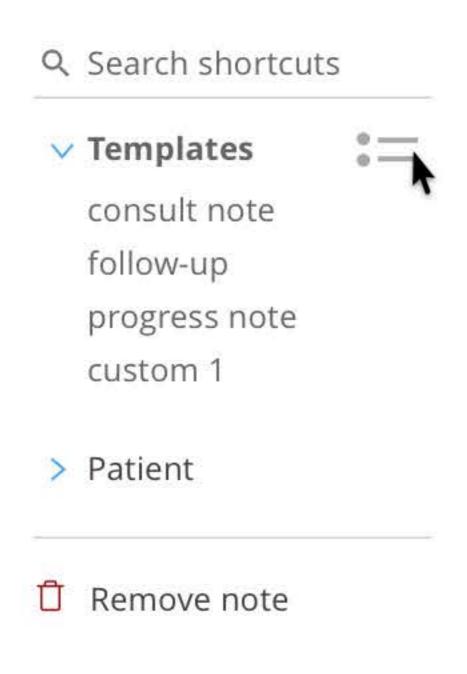
Breast cancer follow-up

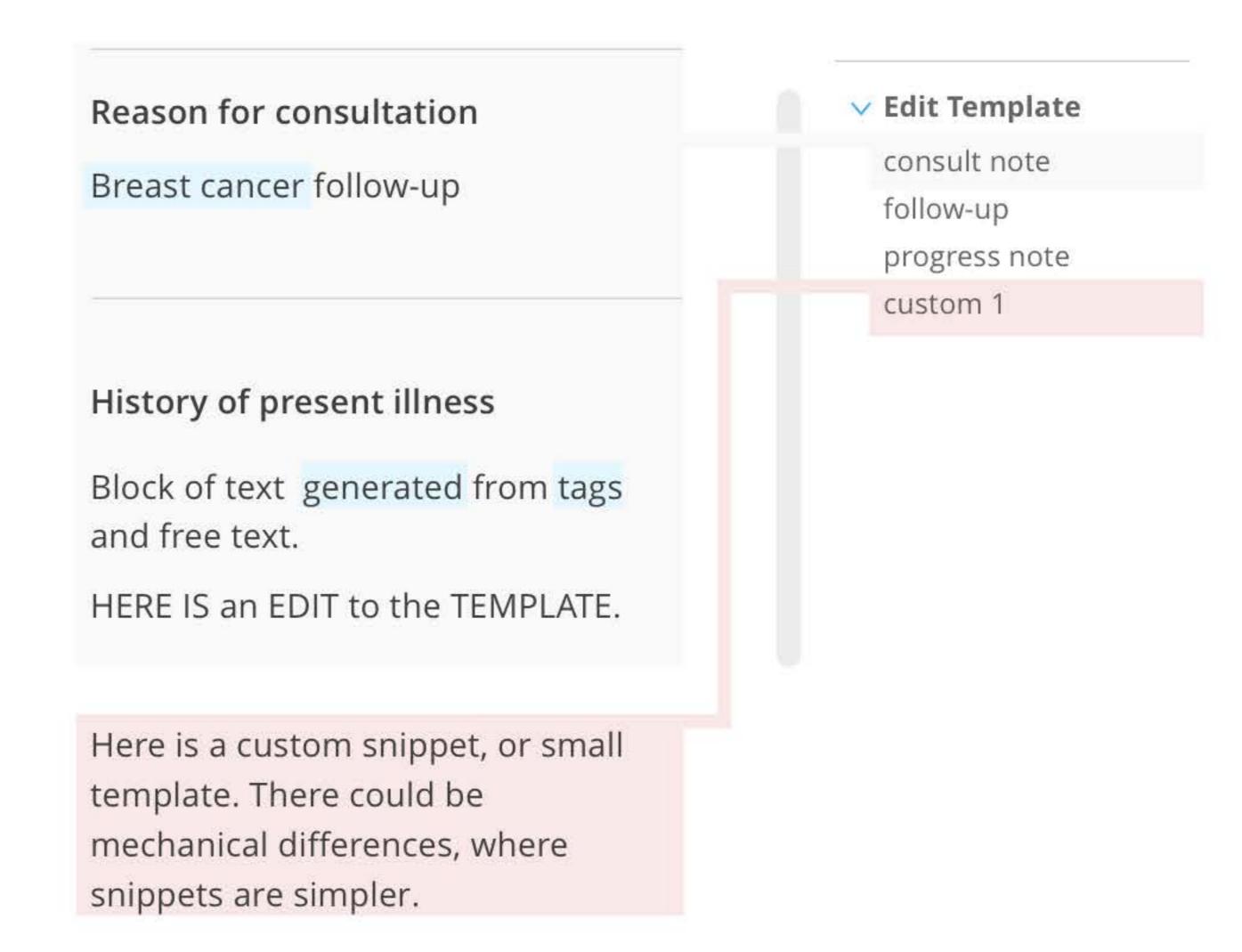
## History of present illness

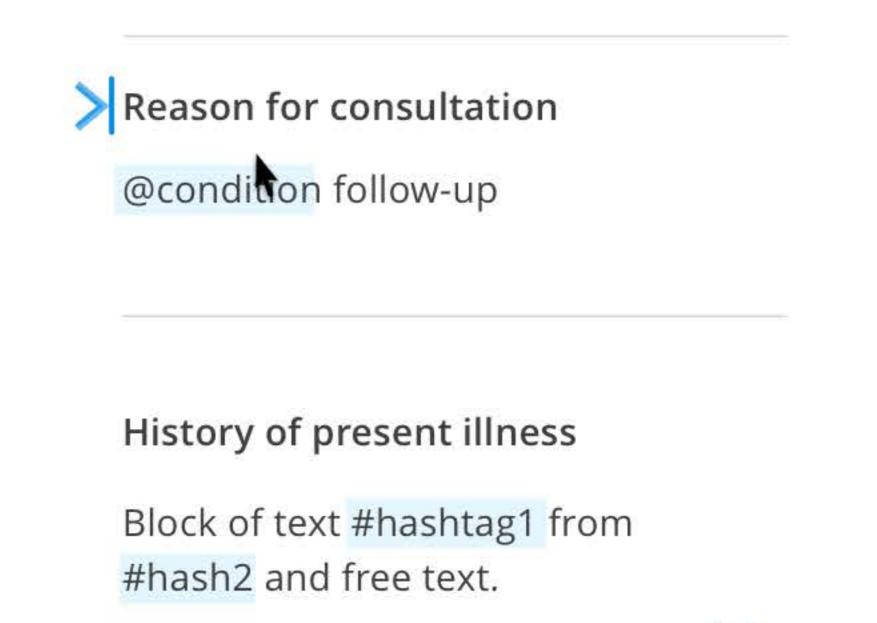
Block of text generated from tags and free text.

HERE IS an EDIT to the TEMPLATE.

Here is a custom snippet, or small template. There could be mechanical differences, where snippets are simpler.







HERE IS an EDIT to the TEMPLATE

Here is a custom snippet, or small

mechanical differences, where

template. There could be

snippets are simpler.

**Edit Template** 

consult note

Available for:

all patients

current only

() Update

× Cancel

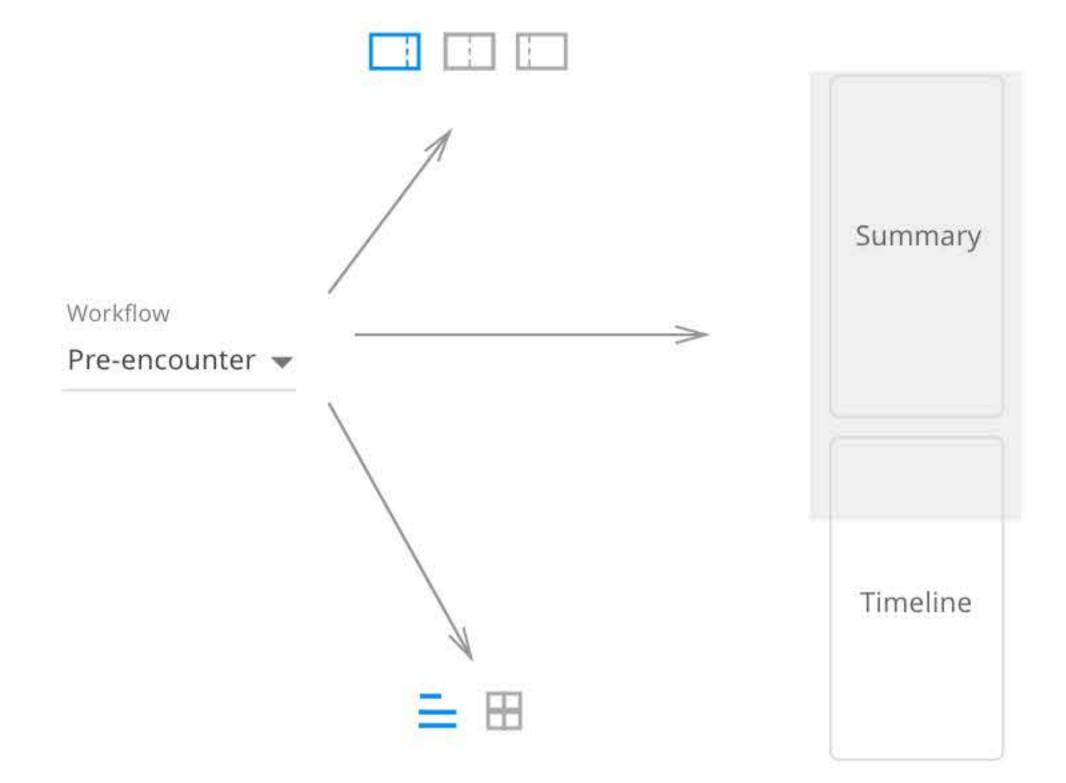
Delete Template

# Custom Workflows

## **Current Situation**

There are three available workflows: pre-encounter, encounter, and post-encounter. Each workflow is characterized by:

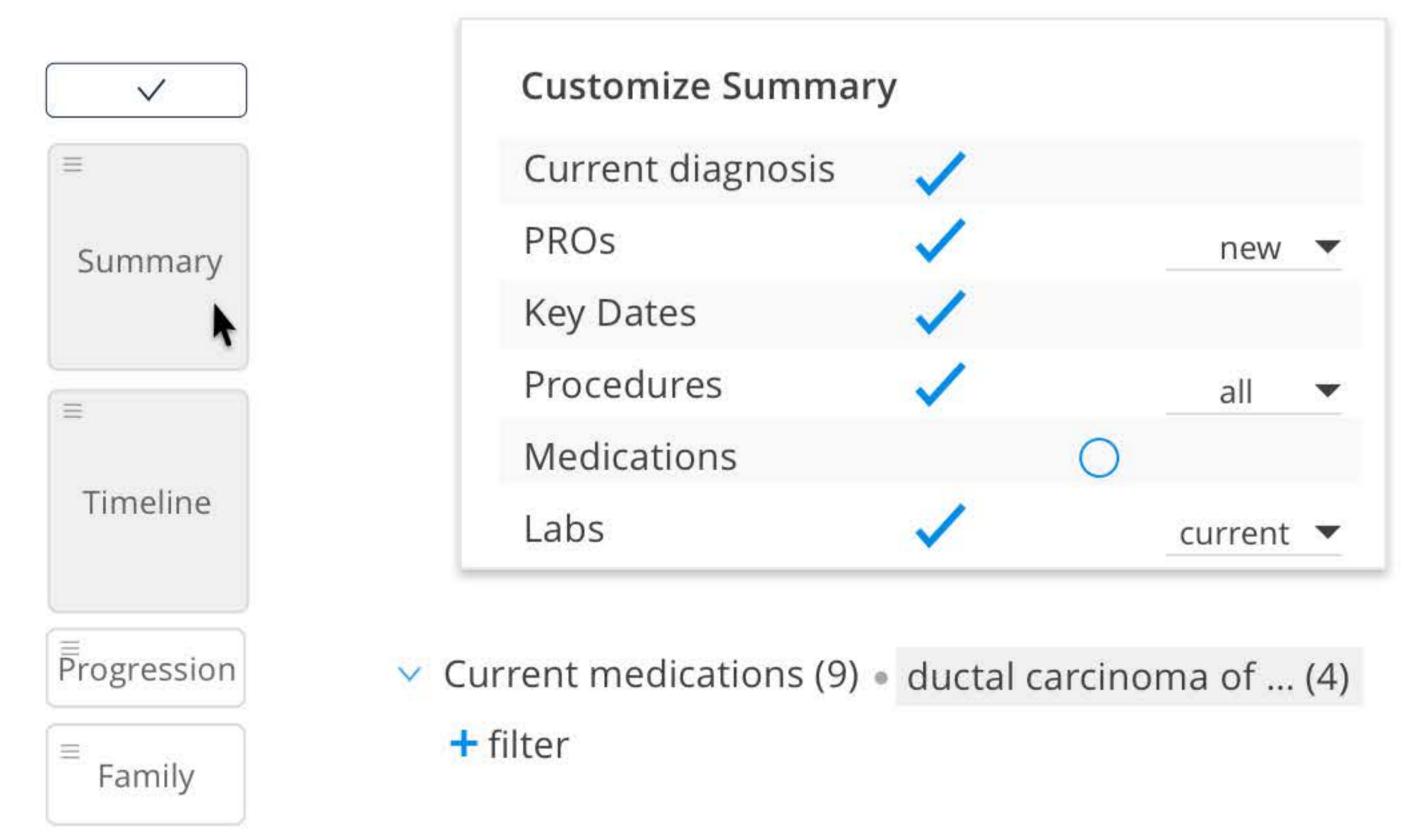
- 1. The view (split of screen)
- 2. Visible sections in mini-map
- 3. Mode of each subsection (narrative, tabular, etc.)
- 4. Arrangement of mini-map



#### **New Customizations**

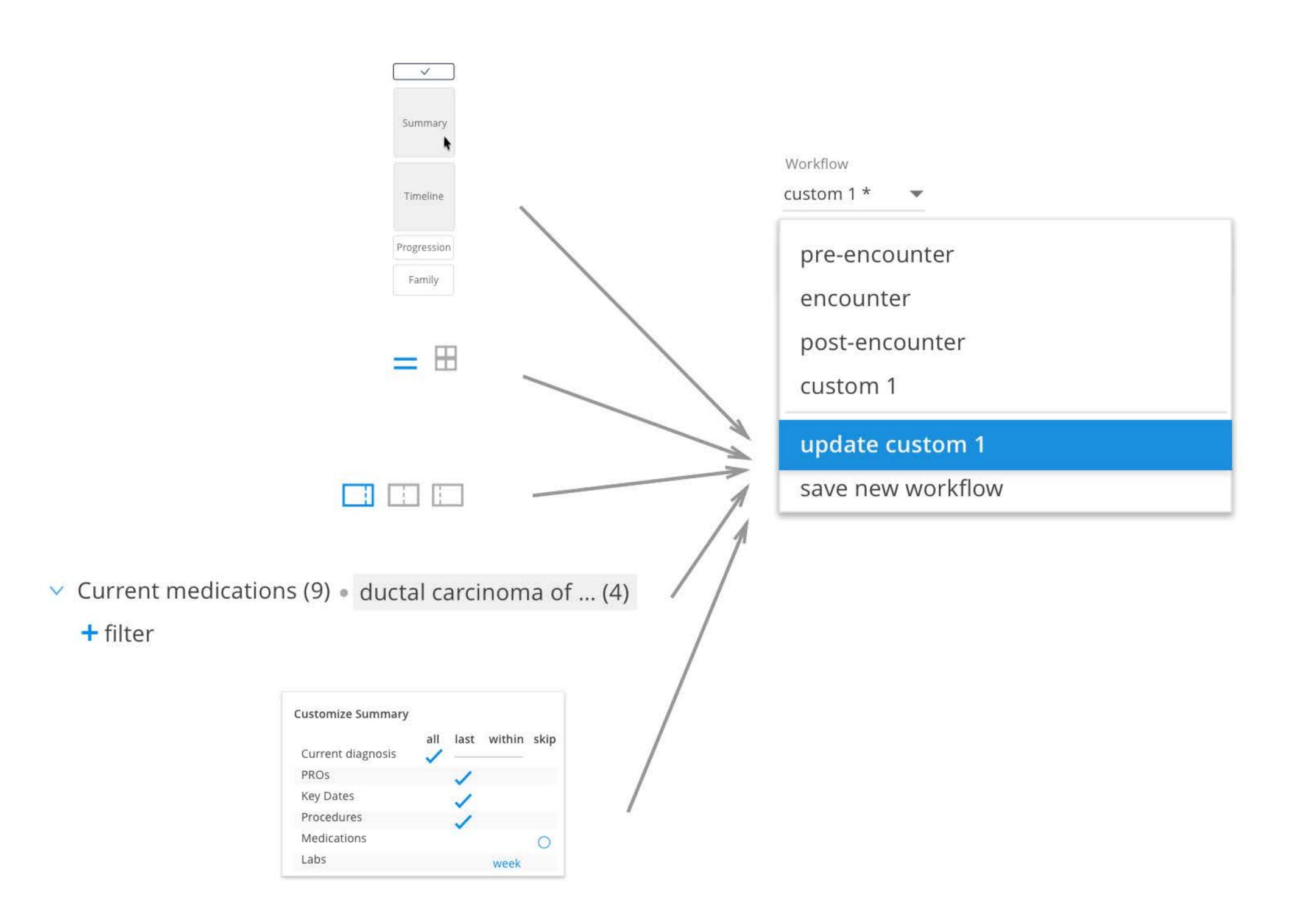
Several elements of the UI lend themselves very nicely to user customization. Some have been explored previously.

- 1. Visiblity and arrangmenet of mini-map
- 2. Templates applied to narrative forms
- 3. Selection of elements for summary
- 4. Filters applied to sections and subsections



# **Saving Workflow**

The set of customizations can be saved as a new workflow, or can update an existing workflow.



# IDE/Editor Inspiration

## **Typechecking**

In an IDE or code editor, function parameters, assignments, comparisons, and return values are validated for data types. Errors are highlighted as the user types.

#### **Function prompt**

When entering a function, helper text often appears that prompts the user with the signature (parameters and return types).

### Layout customization

Editors are highly customizable in terms of window sizes, layout, duplication, tabbing, and tiling.

## Style customization

Editors are highly customizable in terms of coloring and styling, both of syntax and window appearance.

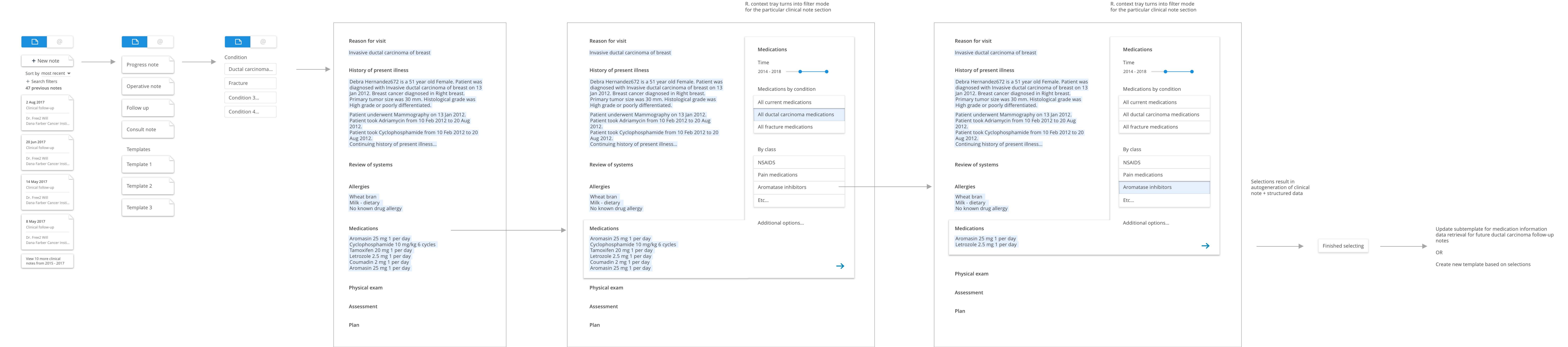
## **GoTo and Searching**

Many IDEs and Editors have searching as well as bookmark/goto functions. Searches are powerful and regex enabled and open persistent windows. Goto functions will let the user quickly access particular sections of the code.

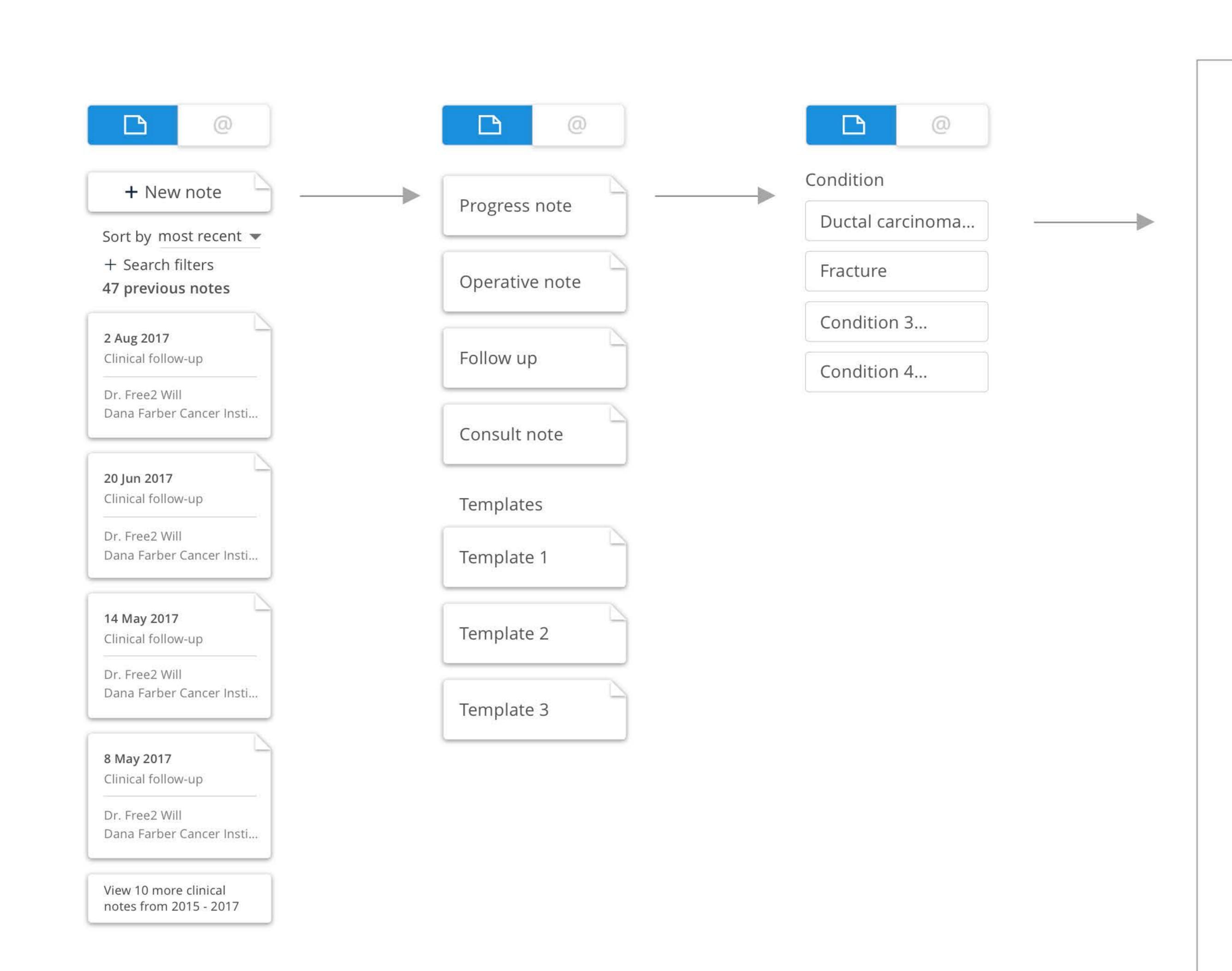
There are a few analogous ideas for FluxNotes:

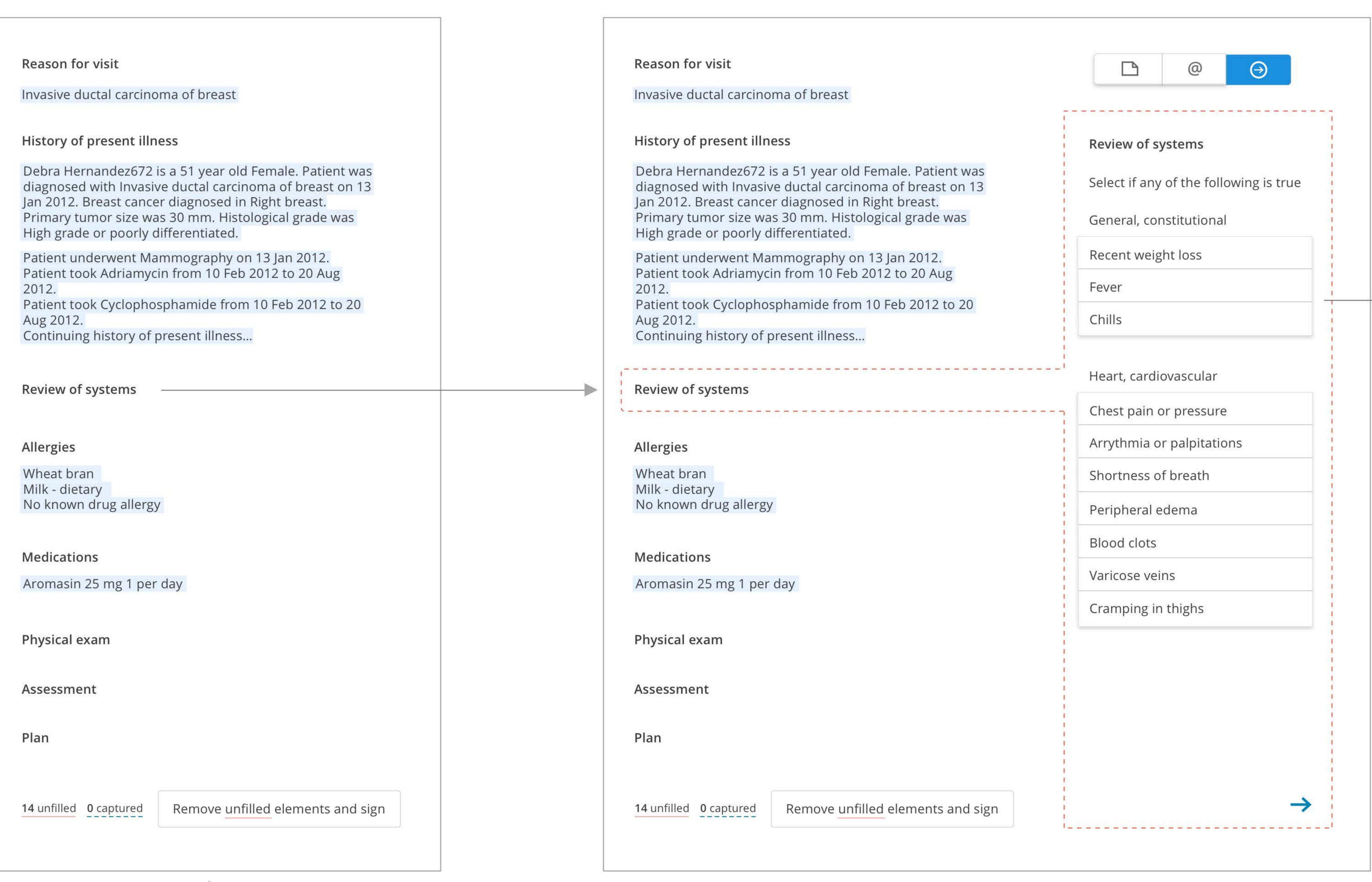
- 1. Captured numeric values can be checked against a reasonable range
- 2. Unit inclusion can be enforced, and those units can be compared against expected.
- 3. As # and @ signs are removed from the interface, advanced part of speech matching can be used to identify and process data captures/references in the notes.
- 4. Spellchecking
- 1. During a capture of reference, the shortcut tree and suggestion dropdown identify possible next texts.
- 2. The dropdown could include whole sentence narrative snippets with blanks for the data. Perhaps the user can tab through the blanks.
- 1. FluxNotes could be flexible and complex as well, with the default workflows presenting very simple clean examples. Typical IDE layouts are complex and lend themselves well to extended immersed work. This is not the use case we are currently working towards with FluxNotes, but there may be some in the future.
- 1. FluxNotes could have several schemes, including dark palettes.
- 2. We currently highlight structured data and missing data in one way. This could be extended.
- 1. We could implement a GoTo shortcut, that searches and navigates to sections and subsections.
- 2. Searching should leave persistent search results, with selectable organization, perhaps on the right bar.

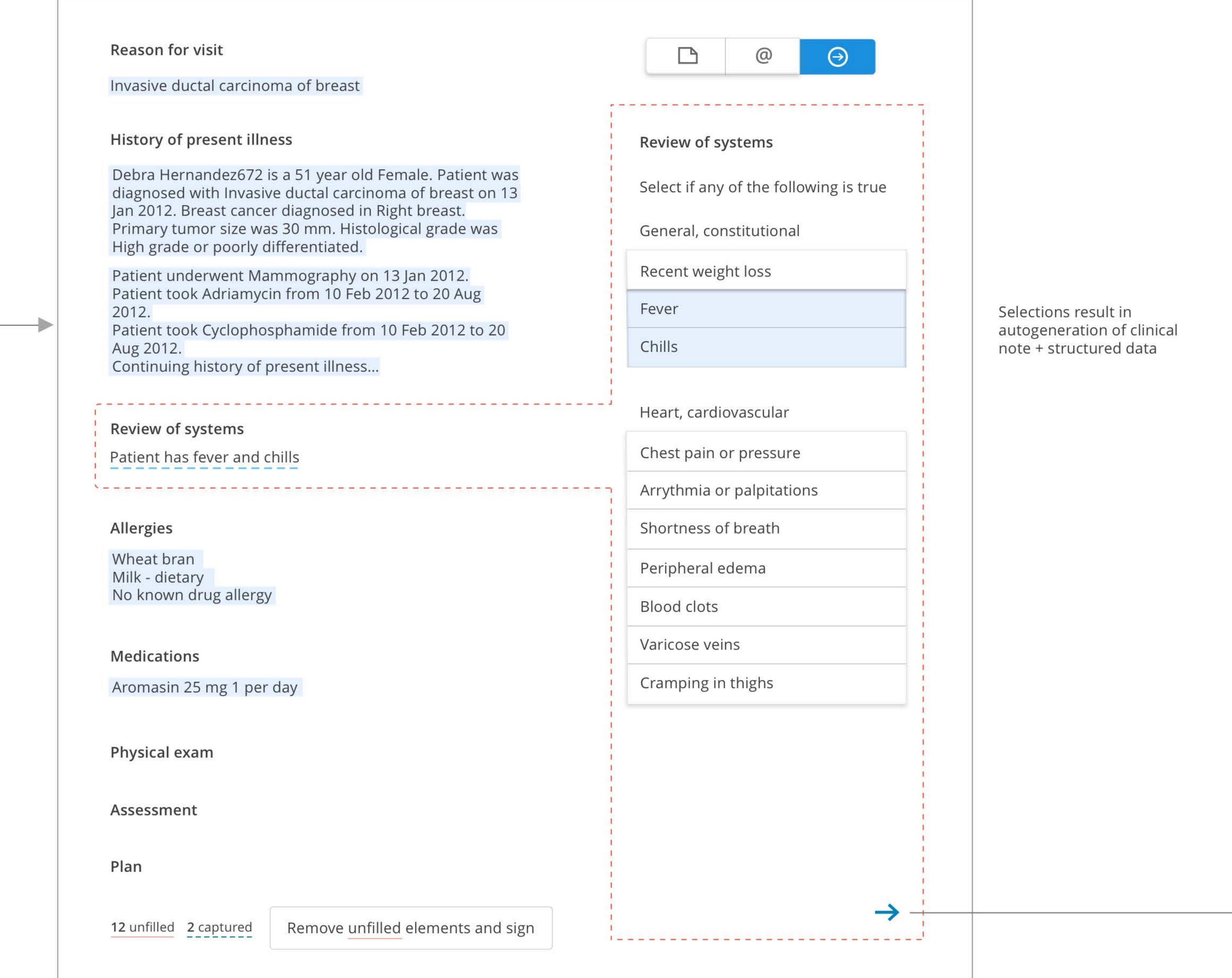
# DATA RETRIEVAL CONCEPT: POS

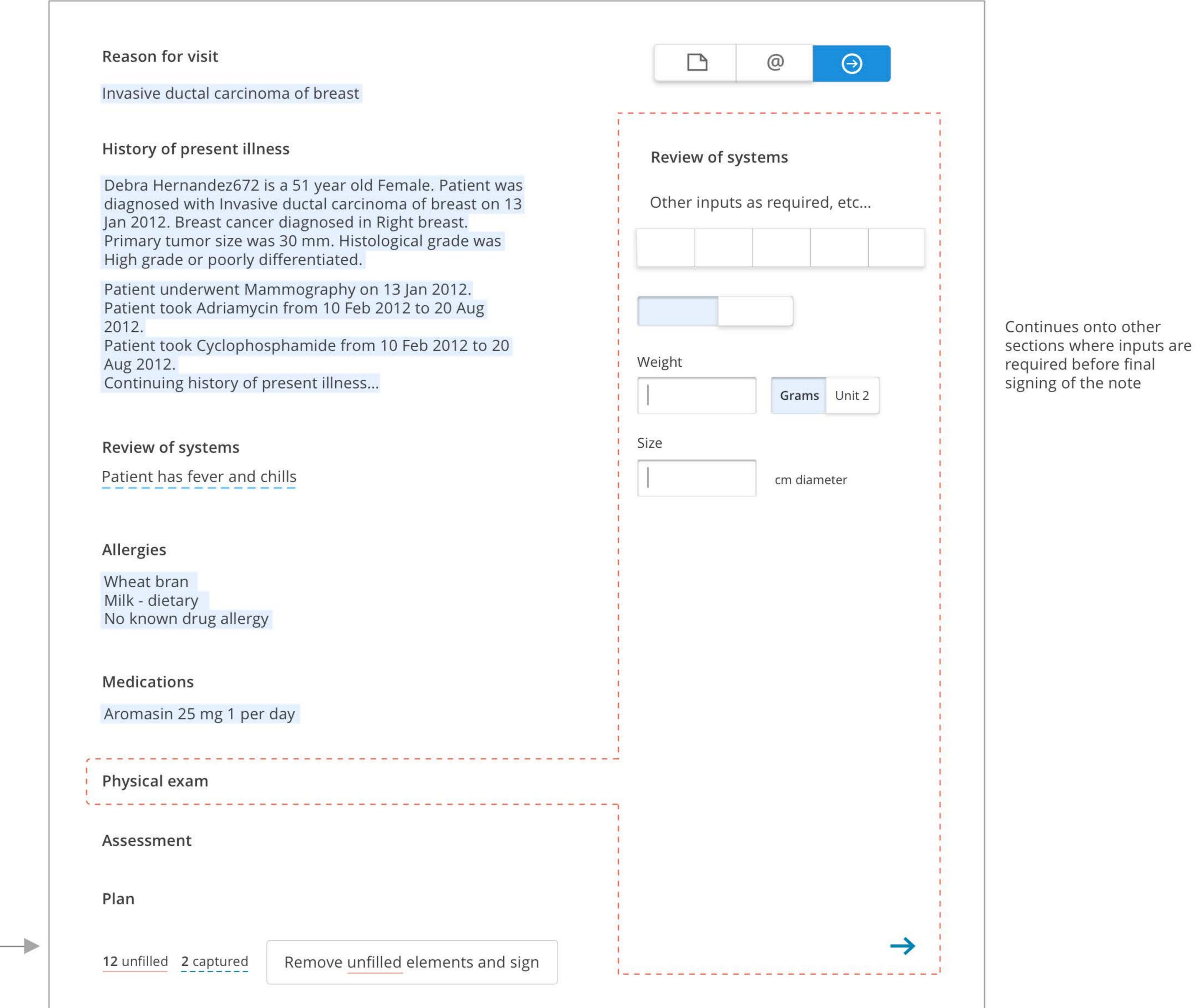


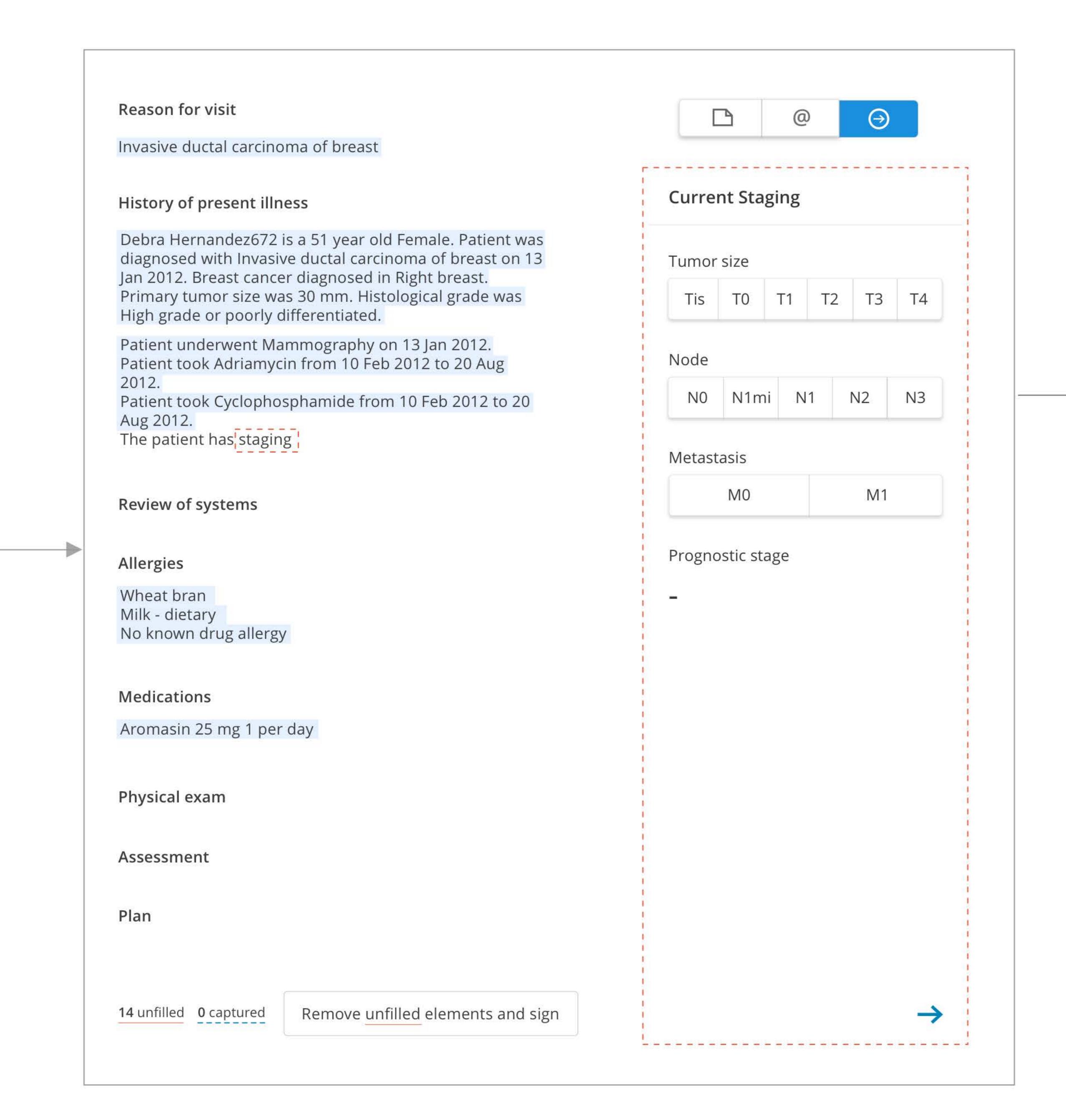
# DATA CAPTURE CONCEPT: WIZARD UI

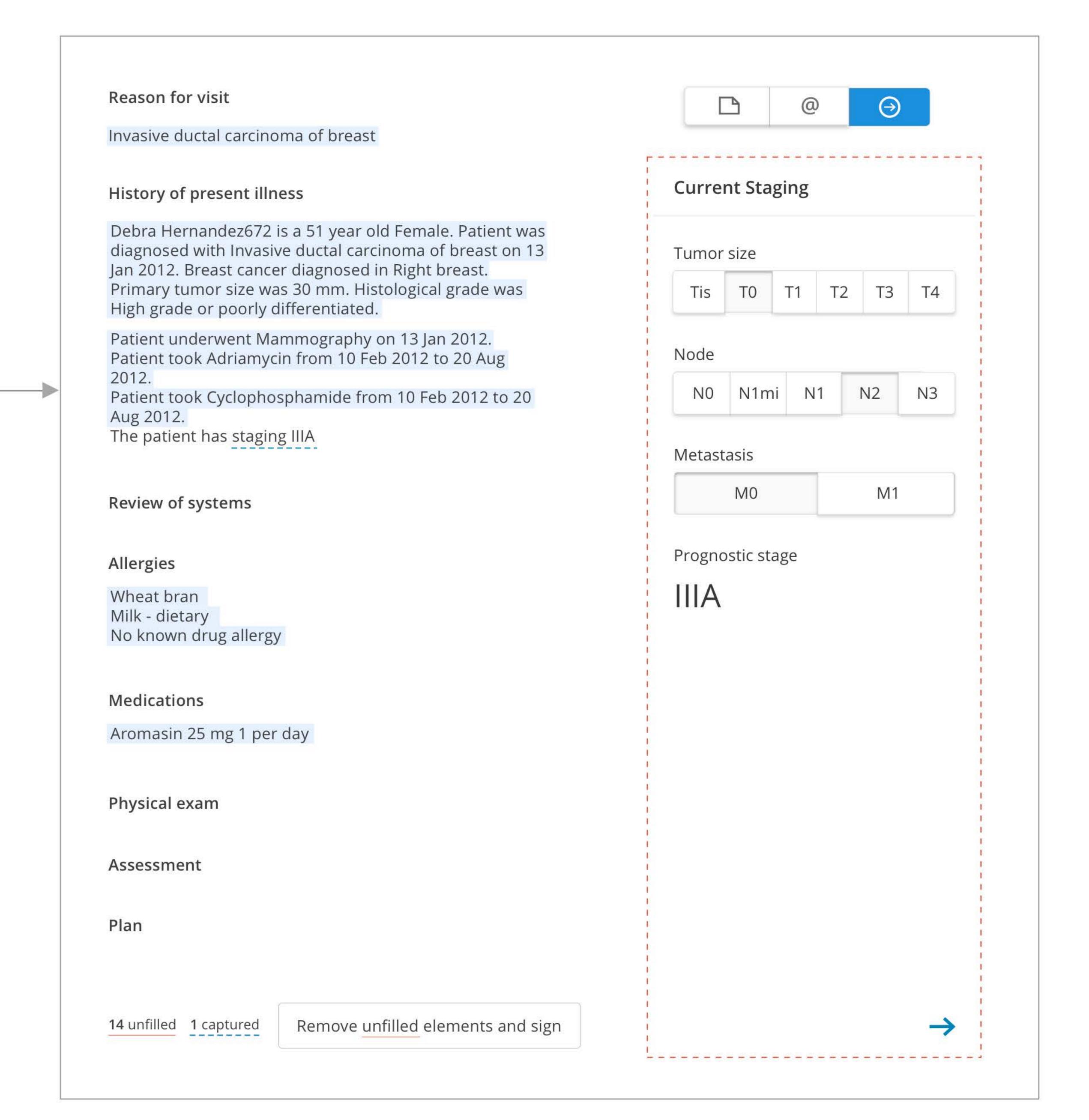












# DATA CAPTURE CONCEPT: ALWAYS ON, "CHECKER" LIST

