Visualization of structured search information

ID: 33650288

Name: Jingfan (Evan) Zang

Abstract

This project tries to create new features based on the 2Dsearch platform. The new tool is aiming to help users quickly understand the inner logic of the existing search strategy. The design workflow contains systematic research on different types of users and an agile design iteration process with building prototypes and multiple round usability testing. In the end, this study would discuss the limitation and the future development direction of the project.

Literature review

Evidence for users need to understand the query string

(1) Errors in the search strategy

Firstly, most of the search strategy templates on the platform are subject to a small number of errors. Meanwhile, there are various types of errors in the search strategy, so the types of errors cannot be simply categorised and automatically fixed by simple algorithms.

(2) The use case for expert searchers

It is important for users to understand the logical structure of the search strategy. Because it gives the user the flexibility to modify the error during use and to change the search strategy according to the search topic.

How could visualization help

The way people understand information visualisation is strongly related to the composition and the cognitive approach of the human brain. the vision of the human brain is pre-attentive due to the sophisticated neural network. This means the visual processes that take place before people pay attention to the things we see.

Progress on 2Dsearch

2Dsearch has already made a lot of progress in visualising the Boolean operator in query strings. It provides the Nested view, the Tree view, and the Inline view for the user to transfer the query string into a 2D graphical image. Three views use different kinds of methods to do the visualisation. After analysis it was found that Nested view had certain advantages in this project.

Design Objective

This project is aiming to use different design methods to solve two questions. (1) Who do I design for? (2) What to design? The two questions are not independent, but their answer influences each other. This project requires finding a balance between two questions.

From the second-hand study. There is still a cognition gap between the text format query string and the 2D canvas image. The opportunities for this project

zang.xueqiu@outlook.com

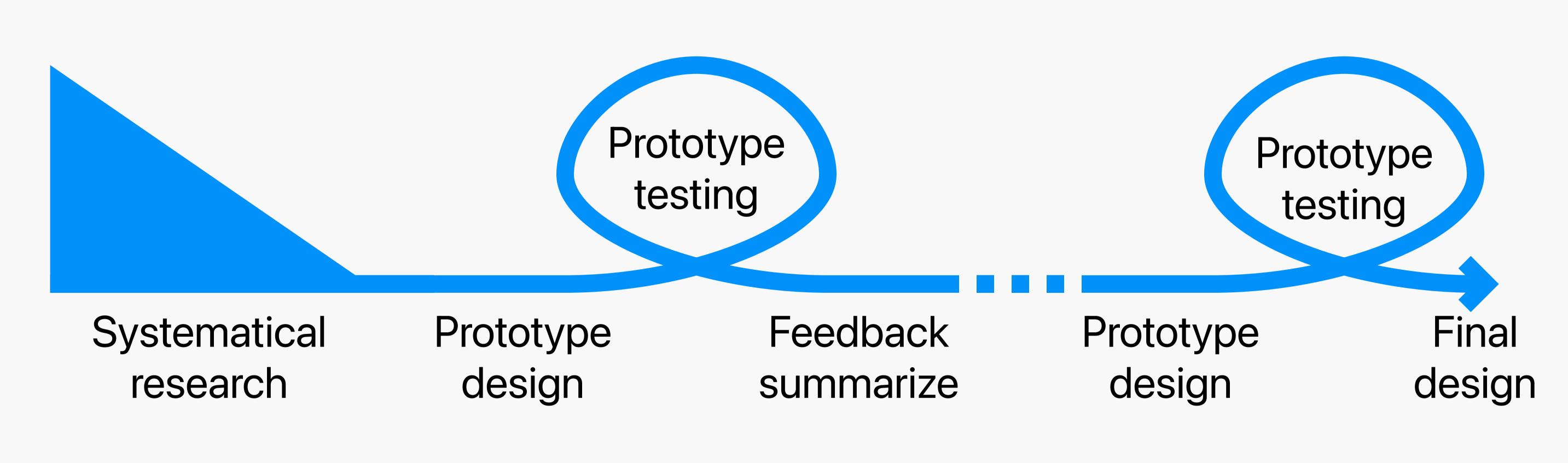
Methodology

Systematicial research

Use the questionnaire to do quantitative research about the working habits of both expert searchers and students. Analysis of questionnaire results to create personas for different types of users.

Agile prototype iteration

There is no static design approach prepared. In the contract, the design process would be dynamic cause this phase would focus on solving the problem that occurs in the prototype.



Stage 1: Questionnaire

Insight for Expert searchers

(1) Proficiency in the concept of structured search. (2) Have lots of experience using other's query string. (3) The main issue is the detailed problem and tech problem.

Insight for Master students

(1) Have difficulty in building their search strategy. (2) No high demand for understanding other's search strategy.

Stage 2: **Prototype pilot testing**

Design objectives for this phase

(1) Create some interface that can be use for testing. (2) Check if the test environment is valid.

Design the prototype

(1) Coding view: Express the structure of the query string better.

- (2) **PLAYER:** Show the generation animation of the query string.
- (3) **DEBUGGER:** Help users to locate the content on the coding view.
- (4) **Different layout:** Explore the information architecture.

1	"Aspergillus"[MeSH]	ch × +						
		app.2Dsearch.com/query		Jery				
2	"Aspergillosis"[MeSH]	Pagy: Unnamed Image: Share Layout: Nested Image: Tree view	Search: Lens.org -	SHARE Layout: Nested - TREE VIEW	Search: Lens.org -			
		Field tag input keyword	RESULTS MESSAGES QUERY DEBUGGER	but keyword	RESULTS MESSAGES QUERY DEBUGGER IPLAYER			
3	"Pulmonary Aspergillosis"[MeSH]				1 "Aspergillus"[MeSH]			
4	aspergill*[tiab]	Field tag input keyword		∧ OR	2 "Aspergillosis"[MeSH]			
			1 "Aspergillus"[MeSH]	OR	3 "Pulmonary Aspergillosis"[MeSH]			
5	fungal infaction[tw]		2 "Aspergillosis"[MeSH]		4 aspergill*[tiab]			



Insight for design the prototype

(1) Coding view is helpful for understanding the search logic. (2) Three different layout is worth testing in the formal test.

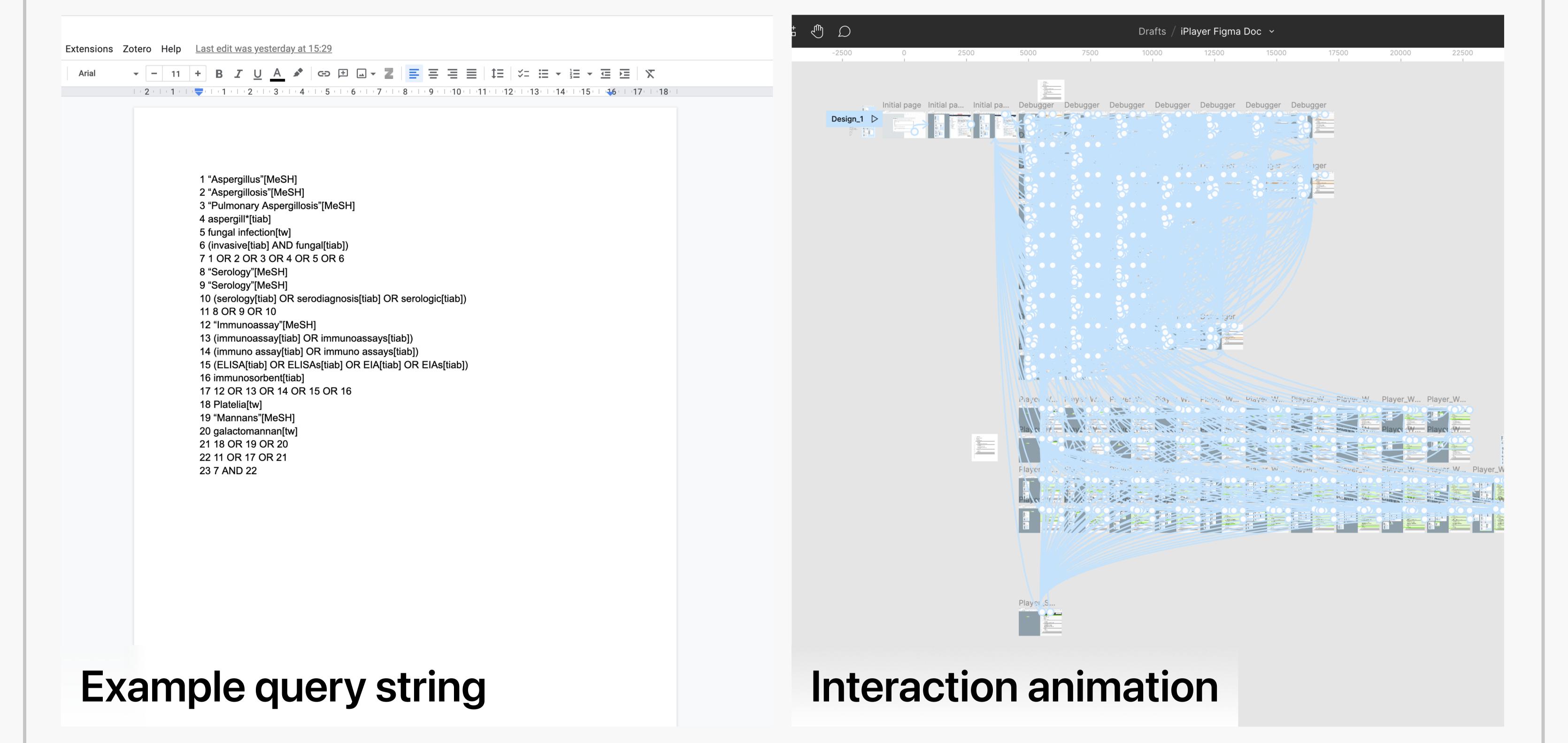
Stage_3: Prototype formal testing 1

Design objectives for this phase

- (1) Test if users can understand PLAYER and DEBUGGER without explanation.
- (2) Verify if debugger and player work in search experts working environment. (3) Testing the layout for PLAYER and DEBUGGER.
- (4) Correct the detail errors that influence the interface user experience.

Design the prototype

- (1) Example query string: Save it in a online text editor.
- (2) PLAYER & DEUBGGER: Add detail and interaction animation.



Insight for persona

- (1) Expert searchers use DEBUGGER in their daily work.
- (2) Expert searchers only use PLAYER for limited situation.
- (3) PLAYER can be extremely useful in the learning process for students who are training to be expert searchers. (New target user detected)

Insight for functional positioning

PLAYER: Designed to help users understand the logic of the query string. **DEBUGGER:** Designed to help users correcting errors from the query string.

Insight for PLAYER

- (1) Participants trying to change the query string in PLAYER.
- (2) It would be better if the PLAYER can process part of the query string.
- (3) The animation effect auto-generated by Figma can be improved.
- (4) The coding view can be enhanced to display the logic structure better.





PROTO_Stage_3



PROTO_Stage_4

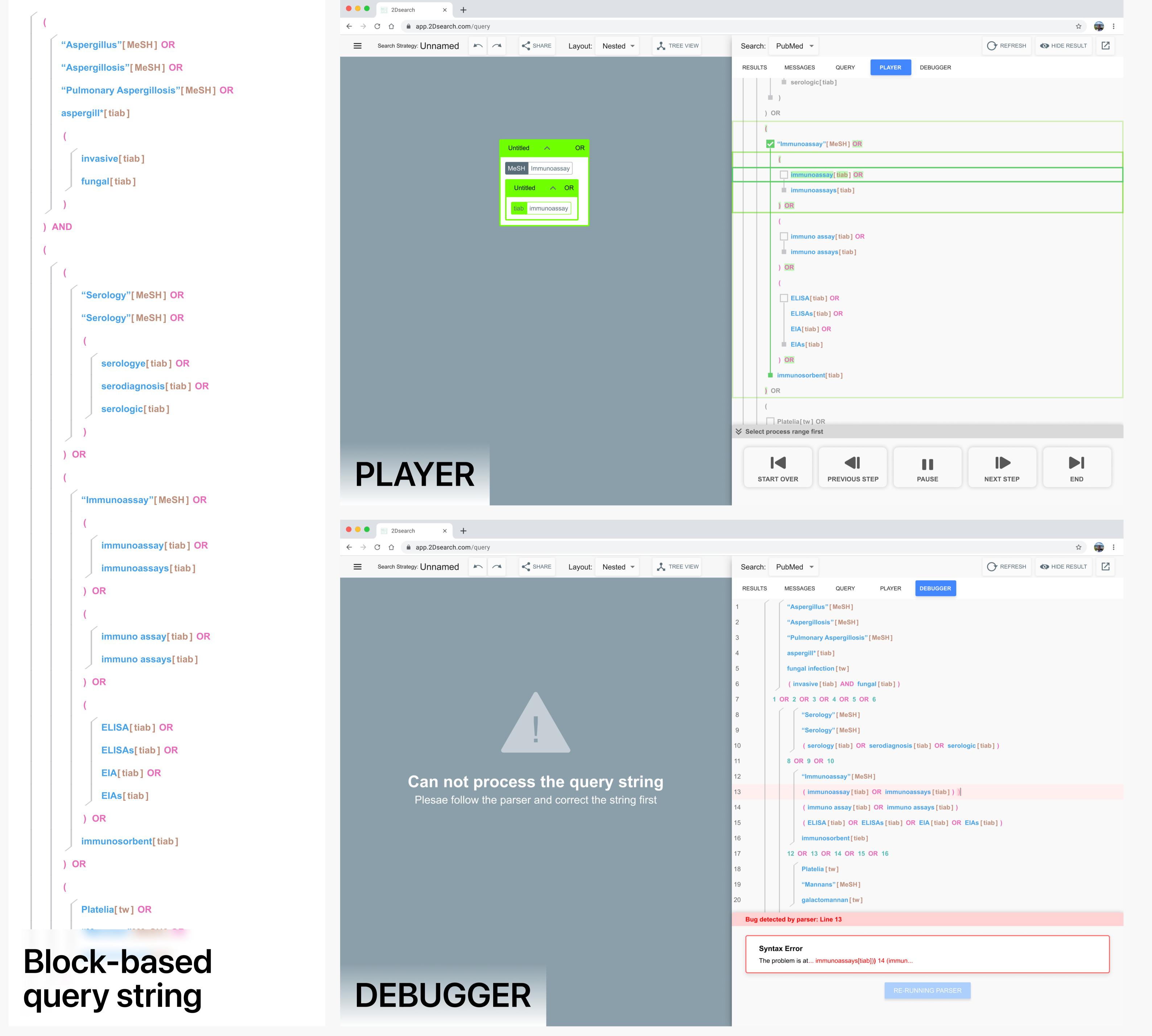
PROTO Stage 2

Design the prototype

(1) New coding view: Bring the colour system into the coding view. Adapting block-based query string and one line query string.

(2) PLAYER: Adding a new feature for processing only part of the query string. (3) **DEBUGGER:** Changing the IA and integrating the parser and message feature all in one DEBUGGER with console.

	Time BUG DETECTED BY PARSER			BUG DETECTED WHILE RUNNING			DEBUGGER_NORMAL					PLAYER_LINE BY LINE			PLAYER_BLOCK BASE			
Component		Default	Cursor on term or field tag	Choosing the bug from console	Default	Cursor on term or field tag	Choosing the bug from console	Default	Click on 2D canvas	Choosing the whole group	Cursor on term or field tag	Cursor on other component	Default	Playing whole query string	Playing part of query string	Default	Playing the line	Playing the block
Coding View	Term	<u>"Keyword"</u> <u>Keyword*</u> <u>Keyword</u>	"Keywo <mark>rd"</mark>	"Keyword"	<u>"Keyword"</u> <u>Keyword*</u> <u>Keyword</u>	"Keywo <mark>rd"</mark>	"Keyword"	"Keyword" Keyword* Keyword	"Keyword"	"Keyword"	"Keywo <mark>rd"</mark>	"Keyword"	"Keyword" Keyword* Keyword	"Keyword"	"Keyword" <mark>"Keyword"</mark> "Keyword"	"Keyword" Keyword* Keyword	"Keyword"	
	Field tag	[Field Tag]	[Field Ta <mark>g</mark>]	[Field Tag]	[Field Tag]	[Field Ta <mark>g</mark>]	[Field Tag]	[Field Tag]	[Field Tag]	[Field Tag]	[Field Ta <mark>g</mark>]	[Field Tag]	[Field Tag]	[Field Tag]	"Keyword" [Field Tag] "Keyword"	[Field Tag]	[Field Tag]	
	Boolean Operator	OR AND NOT	(AND)	(AND)				OR AND NOT		AND AND		AND AND AND	OR AND NOT	AND	"Keyword" AND "Keyword"	OR AND NOT	AND	AND
	ADJ	<u>ADJX</u>	(ADJX)	(ADJX)				ADJX		ADJX		ADJX	ADJX	ADJX	"Keyword" ADJX "Keyword"	ADJX	ADJX	ADJX
	Line	25	25	25	25	25	25	25		25		25	25	25	"Keyword" 25 "Keyword"	25	25	25
	Term				input keyword Field tag input keyword Field tag input keyword	input keyword Field tag input keyword Field tag input keyword	input keyword Field tag input keyword Field tag input keyword	input keyword Field tag input keyword	input keyword Field tag input keyword	input keyword Field tag input keyword	input keyword Field tag input keyword							
2D canvas	Group							Unamer A 192		Unumed A		Unamel A	Umumed in . AND	Lourant e 300	Unamed 🔨 — ALE	Umperant ALC	Linux and AND	Unament A AND
Сс	oding	y vie	W C(olou	r sys	stem		ADJ3		ADJ3		ADJ3	ADJ3	ADJ3	AD J3	ADJ3	ADJ3	ELCA



Insight for the prototype

- (1) Colour system enhance the visualization ability to deliver the logic structure
- (2) Block-based query string is easy to understand

(1) The viewalization for array and he may a stand out

(3) New features for PLAYER and DEBUGGER is easy to understand