

[https://drive.google.com/drive/folders/1tFF5LGyDoisKiTATabklcrauzx1\\_fS2j?q=sharedwith:public%20parent:1tFF5LGyDoisKiTATabklcrauzx1\\_fS2j](https://drive.google.com/drive/folders/1tFF5LGyDoisKiTATabklcrauzx1_fS2j?q=sharedwith:public%20parent:1tFF5LGyDoisKiTATabklcrauzx1_fS2j)

# Void's Concrete Architecture

Group 30

Ibrahim Kettaneh ([ibrahim.kettaneh@queensu.ca](mailto:ibrahim.kettaneh@queensu.ca))

Sophie Liang ([22whr@queensu.ca](mailto:22whr@queensu.ca)) Presenter 2

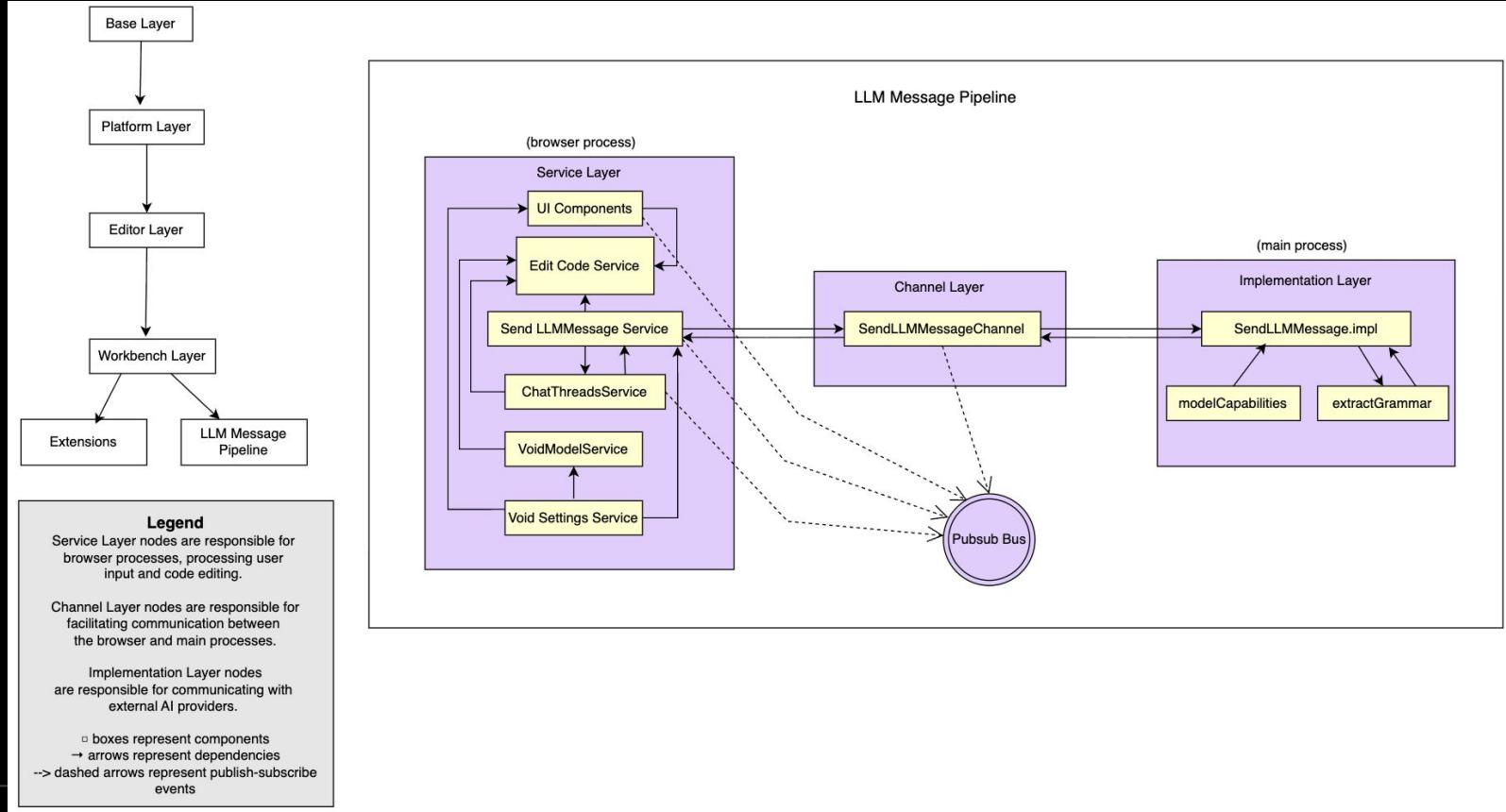
Noelle Morley ([25gdb@queensu.ca](mailto:25gdb@queensu.ca))

Annika Tran ([23LM5@queensu.ca](mailto:23LM5@queensu.ca)) Presenter 1

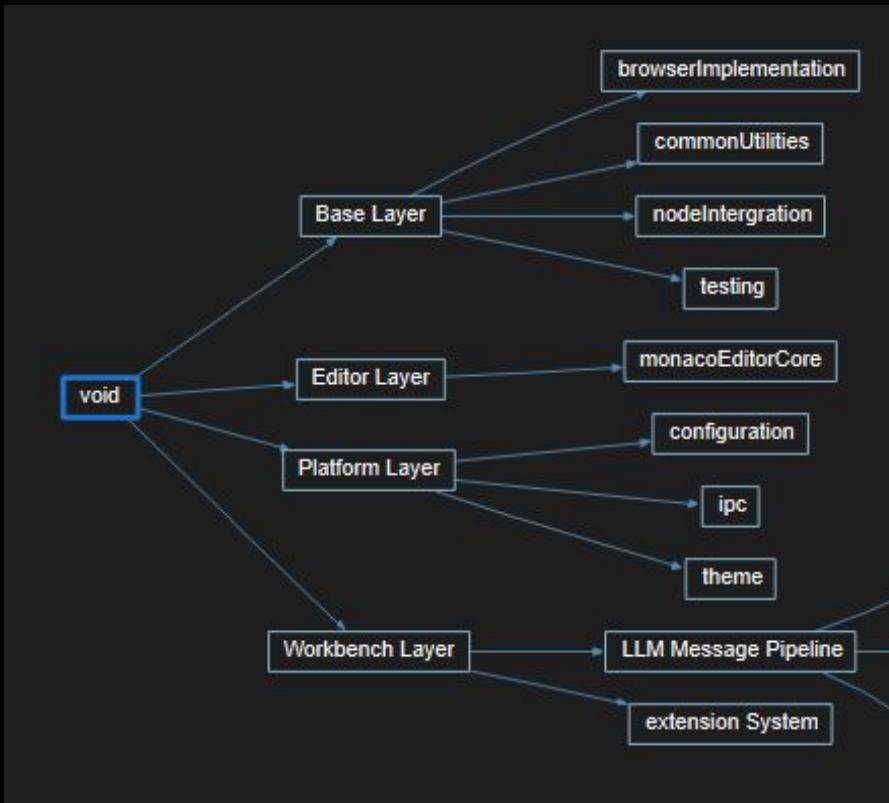
Nicole Wu([22lI20@queensu.ca](mailto:22lI20@queensu.ca)) Group Leader

Joshua Zheng ([23SBNI@queensu.ca](mailto:23SBNI@queensu.ca))

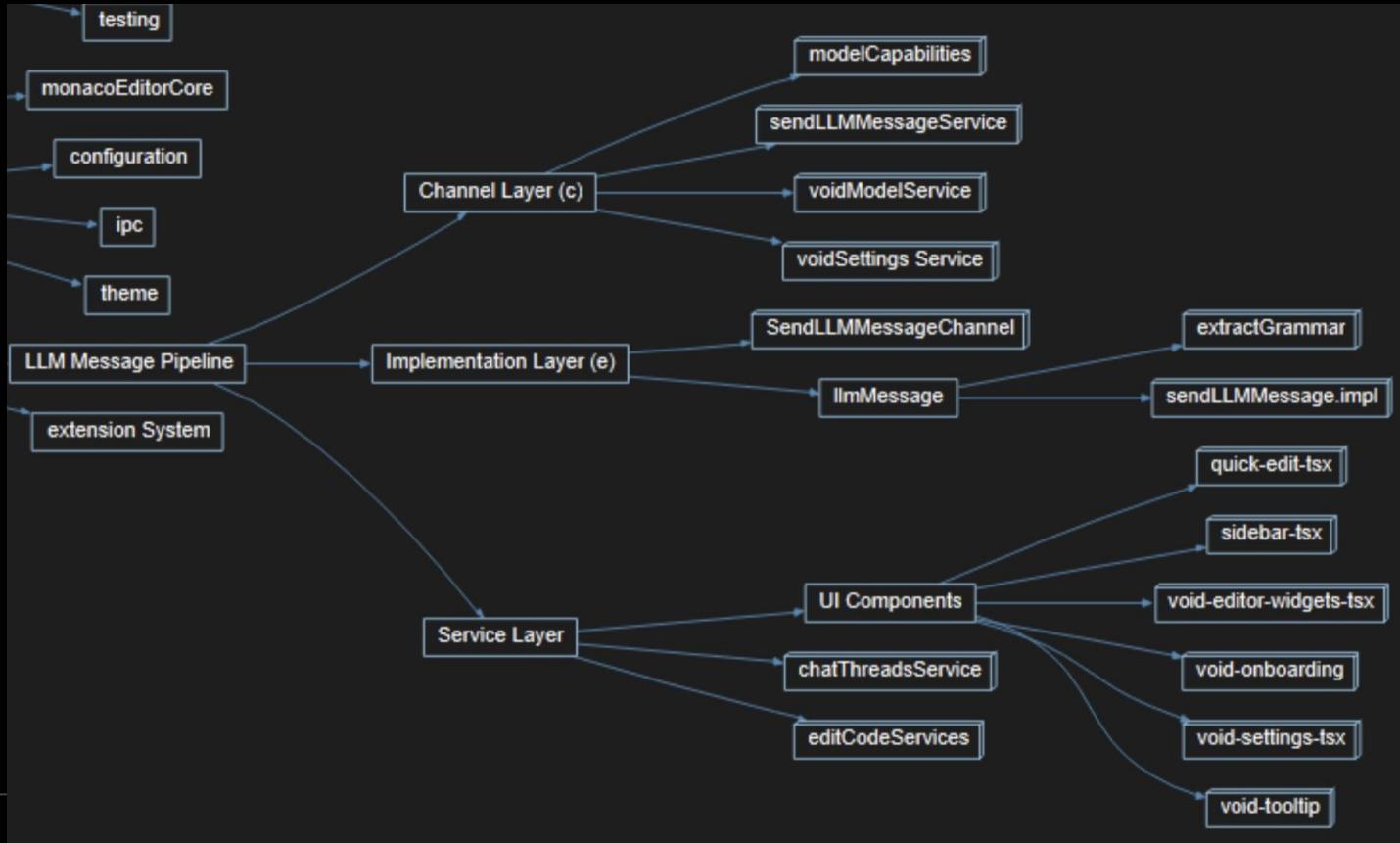
# Updated Conceptual Architecture

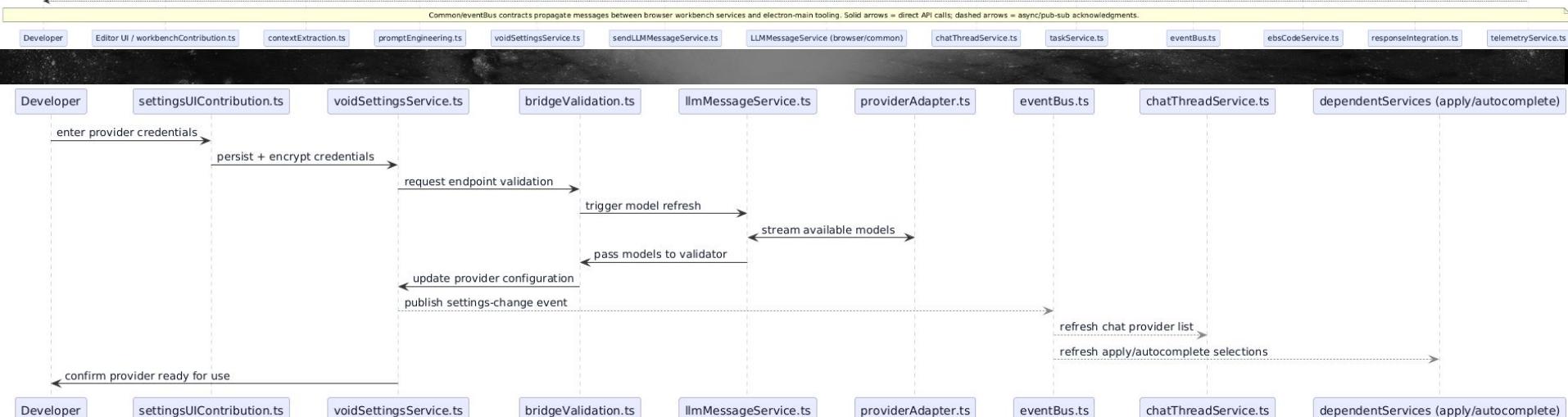
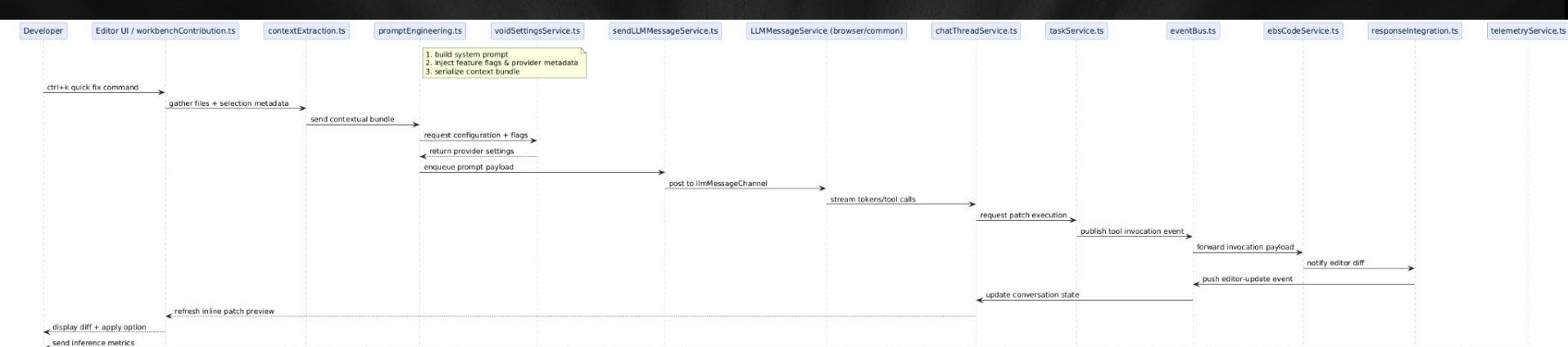


# High-level concrete architecture



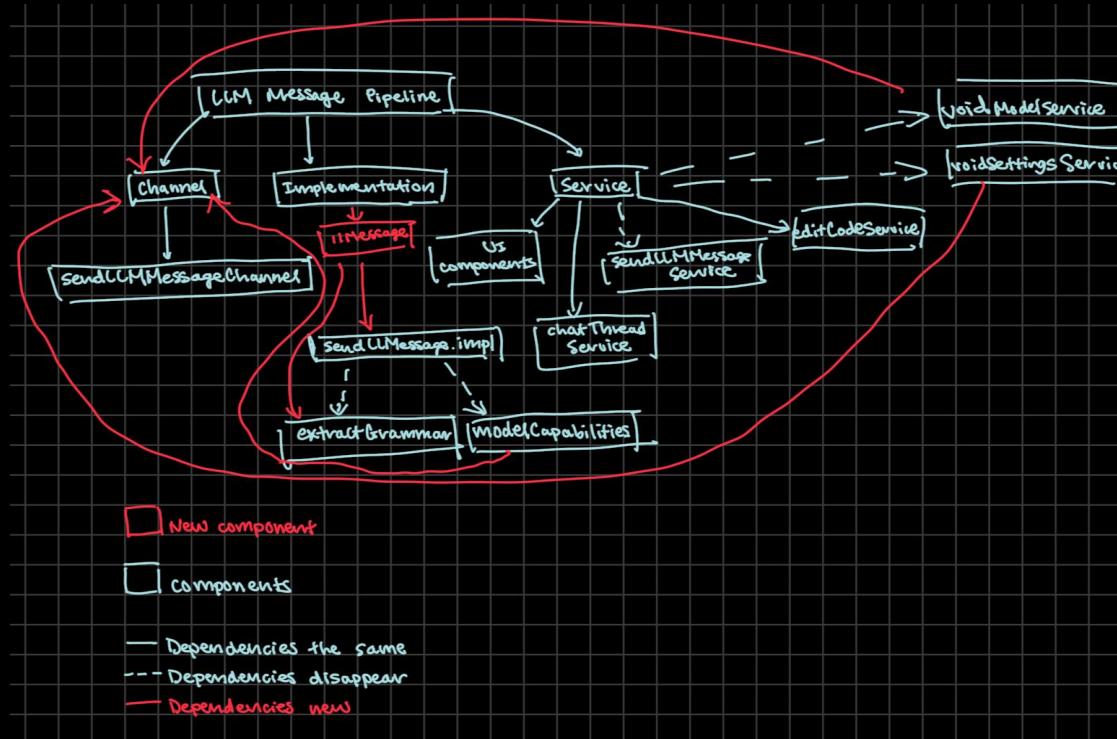
# 2nd level Subsystem





Solid arrows = synchronous configuration or validation calls  
Dashed arrows = Event Bus publish/subscribe notifications  
Double-headed arrows = streamed model responses from provider adapters

# Divergence



# Concurrency and alternatives

- Pub/Sub enables simultaneous processes
  - Layered design allows a team to focus on a layer
- 
- Client/Server Architecture:
    - reduces resources on client
    - increases latency to network communication
    - security risks when data is sent to remote servers
  - Object-Oriented Architecture:
    - reusing and simplifies unit testing
    - causes tight coupling between layers



# Lessons Learned and Limitations

- Learned to use Understand and how to analyze the dependencies
- Found differences between concrete and conceptual

## Limitations

- Understand was incredibly laggy
- Some members could not run it and frequent crashes
- Team had to have full communication to ensure consistency
- Hard to get accurate view without being the actual developers of Void
- 



# AI Teammate

## AI Member Profile and Selection Process

- chose ChatGPT based on accessibility and ease of use

## Tasks Assigned to the AI Teammate

- Proofreading and summarize code

## Interaction Protocol and Prompting Strategy

- Have model open for prompt response

## Validation and Quality Control Procedures

- Limited use, ensure cross reference

## Reflection on Human-AI Team Dynamics

- Inconsistent, but supportive

