BookLeaf

Tools Setup Report

Version 1.0

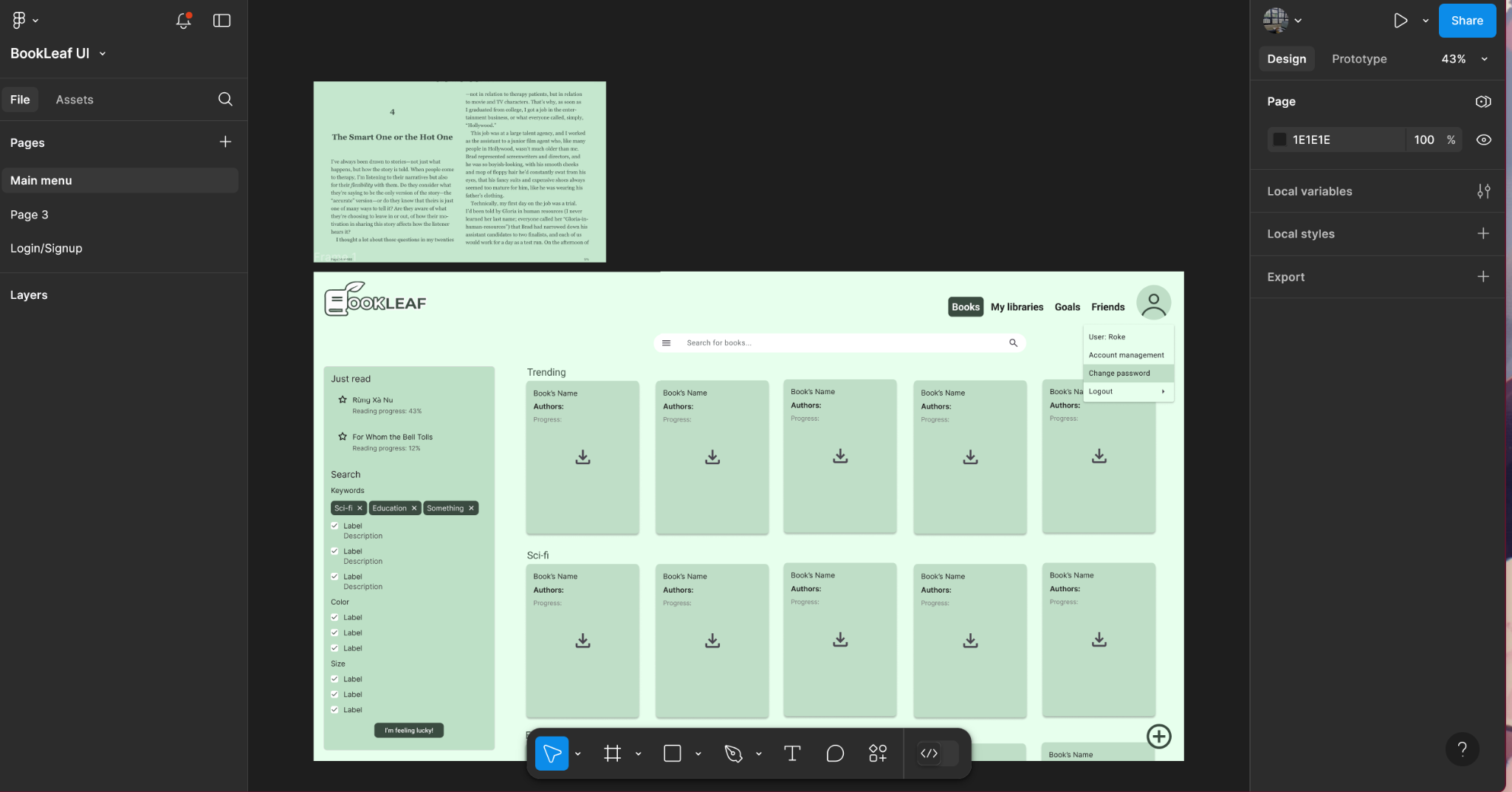
Tools Setup

# Introduction

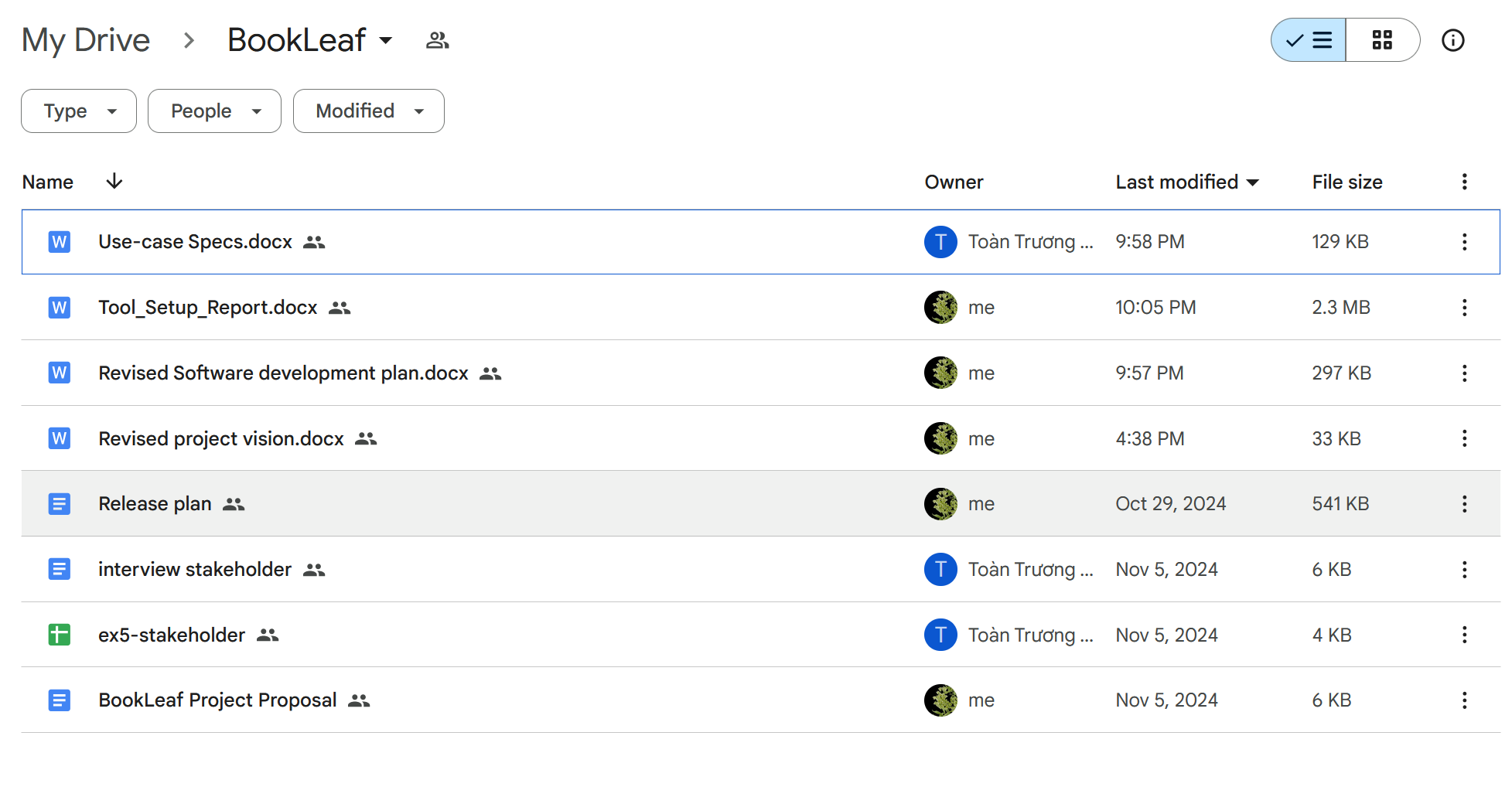
The implementation of appropriate development tools is crucial for successful software project execution and effective team collaboration. This report outlines the essential tools setup required for our team's software development process, including project management, communication, and version control systems. Proper tool selection and setup will serve as the foundation for maintaining organized documentation, coordinating team efforts, and delivering high-quality software deliverables.

# Tools

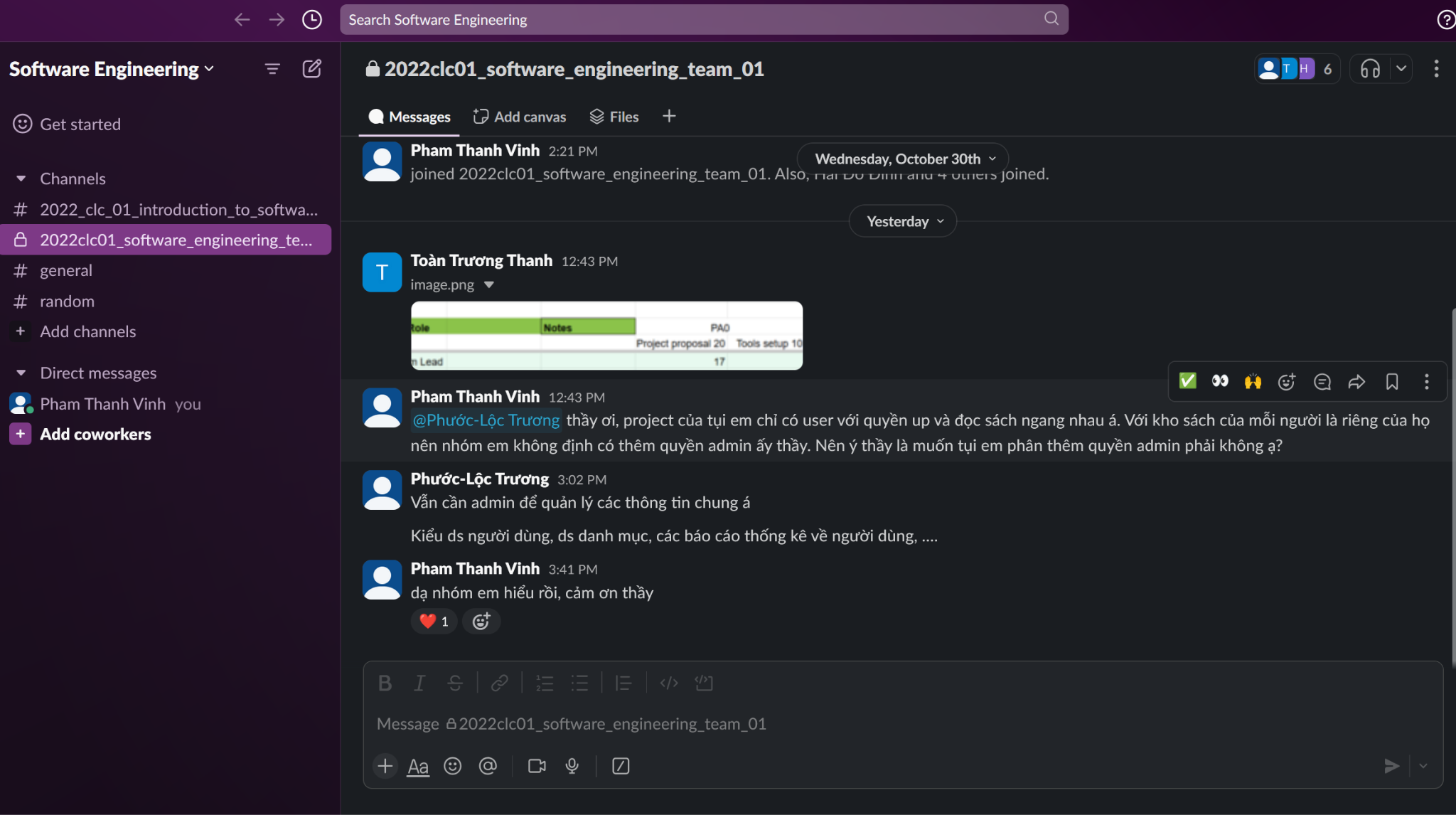
* 1. Moodle
* Used for posting and submitting assignments.
  1. Facebook
* Used for receiving general notifications, class discussions, and questions, answers.
  1. Figma
* Use for UI/UX design of the application.



* 1. Google Drive
* Use for storing documents needed for submission.

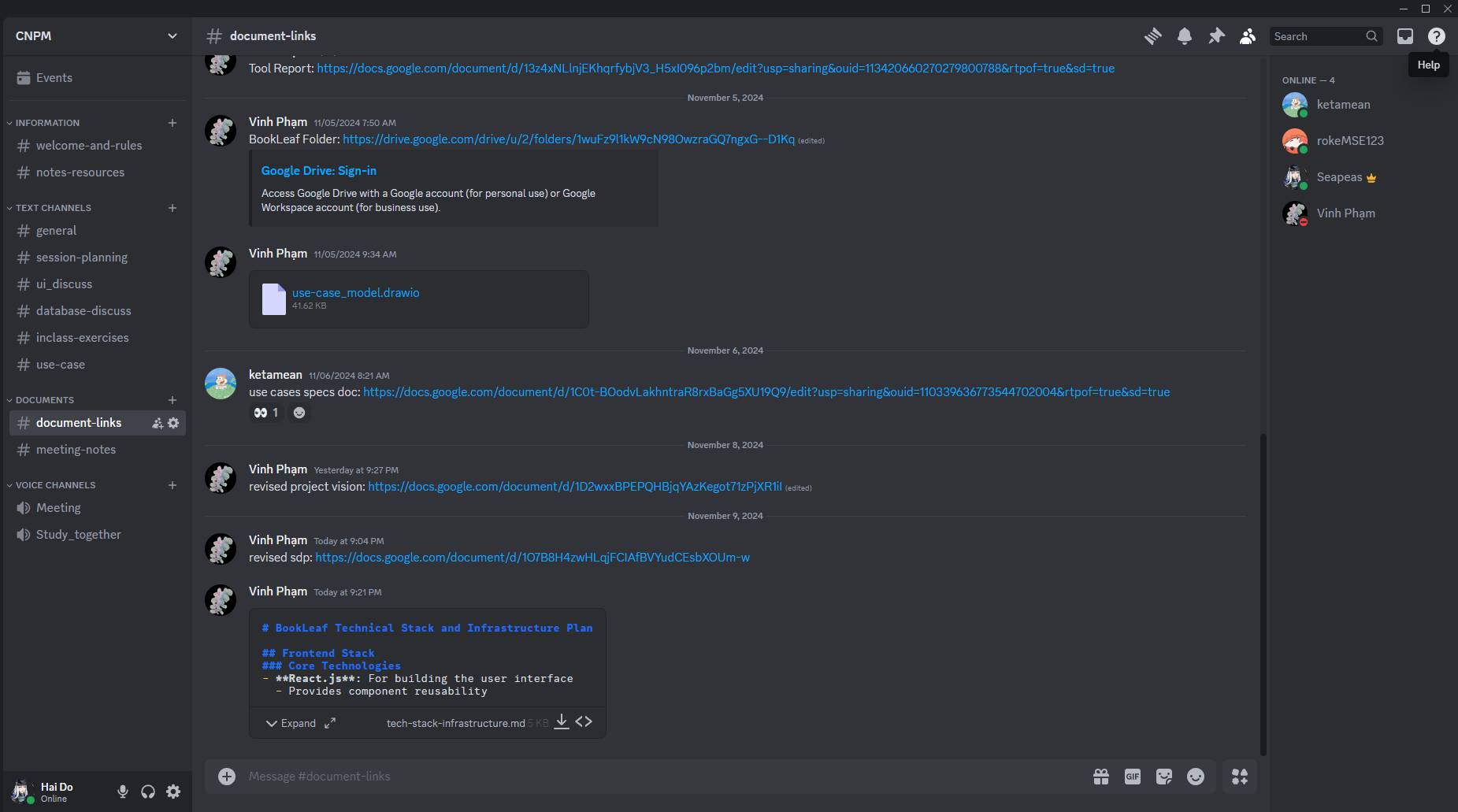


* 1. Slack

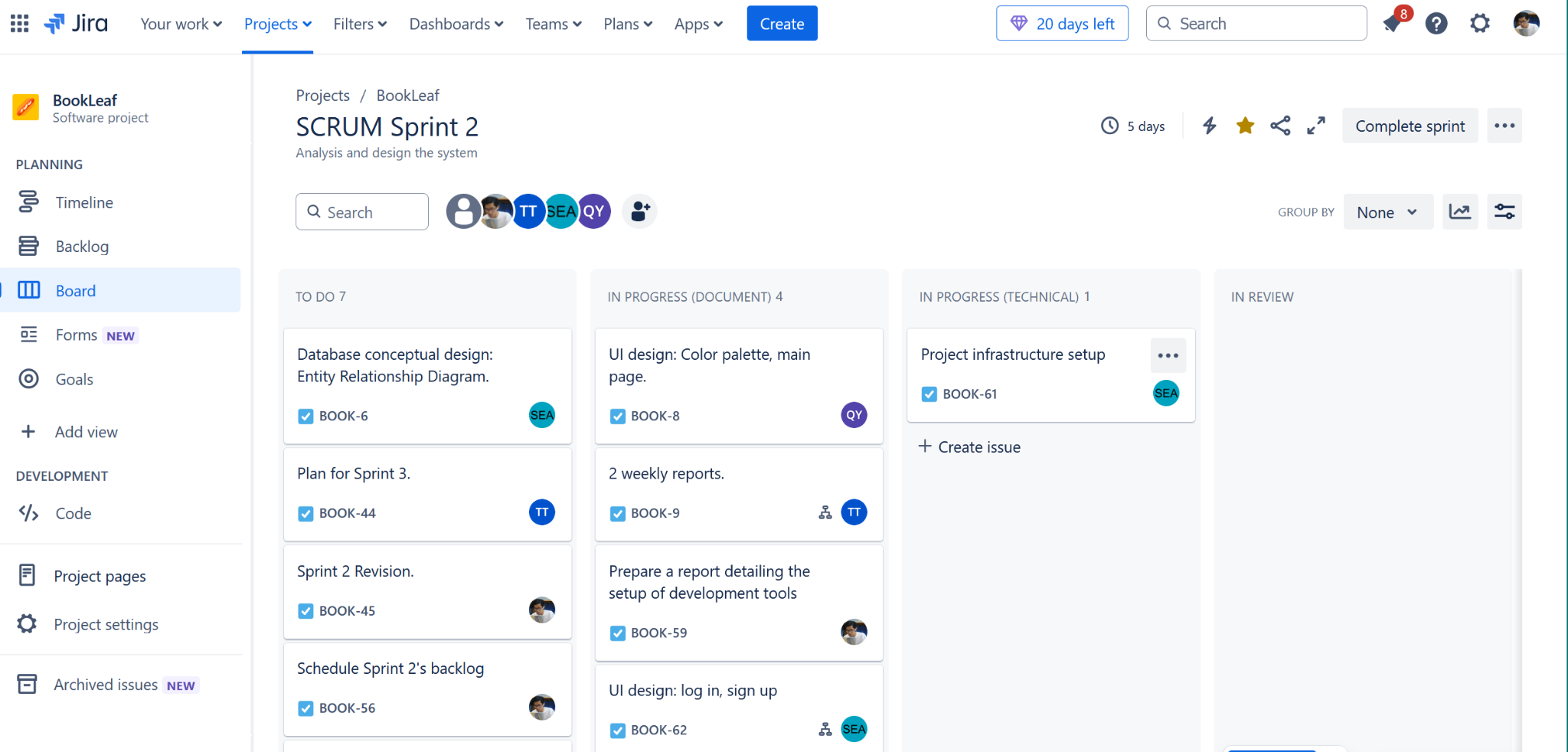


Used for discussions and interactions with teachers and TAs.

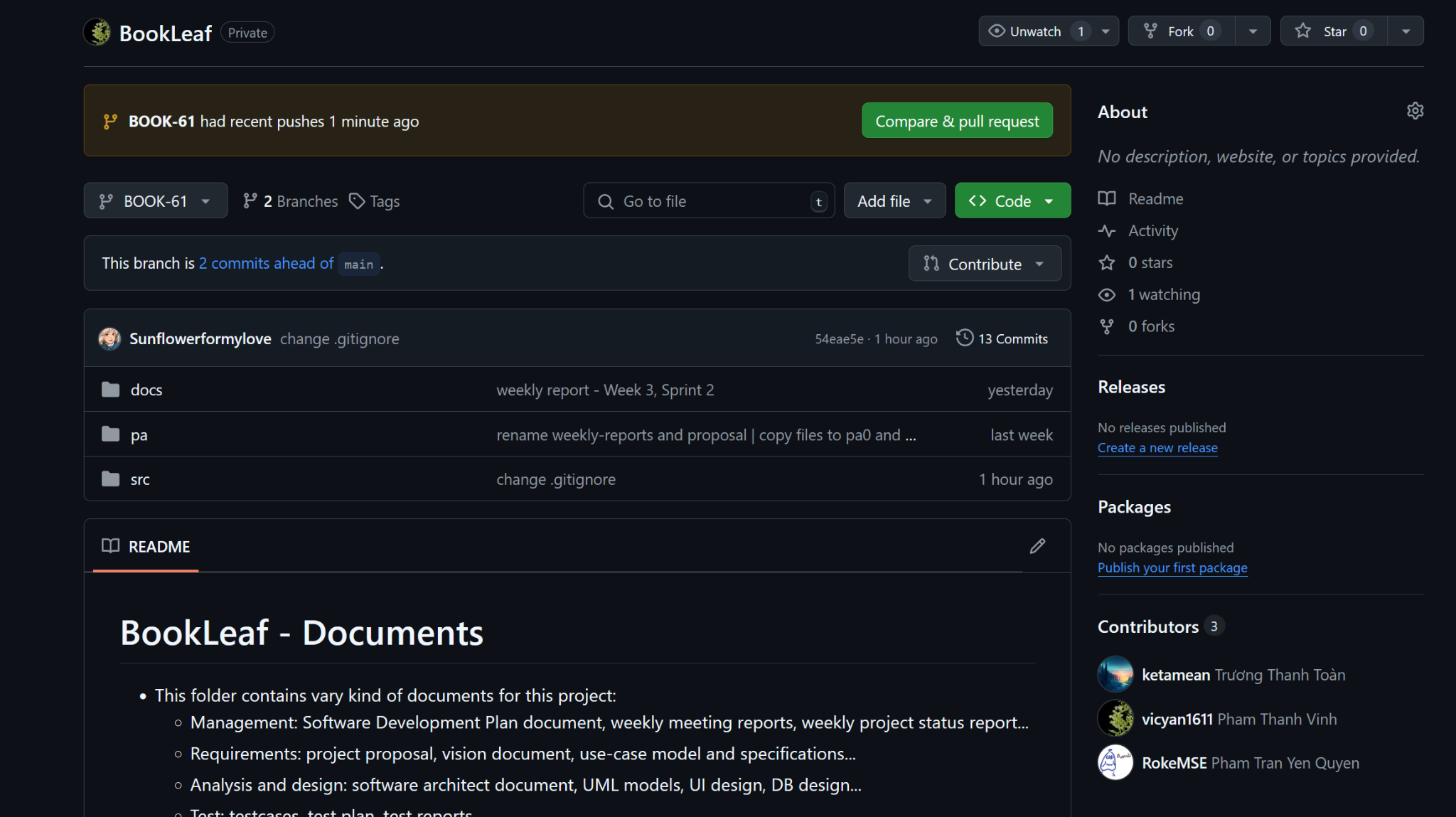
* 1. Discord



* Main communication hub for team collaboration.
* Text channels:
  + *#general:* use for general chatting and discussions. Also works as a place for resolving work-related issues and conflict, democracy voting, etc.
  + *#session-planning:* for planning and discussing sessions.
  + *#ui\_discuss:* for those who are involved in designing, and creating UI/UX can discuss current problems, opportunities and challenges they faced. Also for sharing tools, and templates related to UI/UX designing, as well as resolving conflicts.
  + *#database-discuss:* for those who are involved in the process of designing the app’s databases, either SQL or NoSQL. Mainly for sharing enhanced ERDs and relational diagrams, as well as acting as a playground for discussing the problems, opportunities and challenges faced when designing the application’s database.
  + *#inclass-exercises*: for discussing, contributing, and sharing tools, files, templates, and solutions with regards to the activities which take place in class.\
  + *#use-case:* for those who are involved in the process of designing, creating and discussing the use case diagram and use case specifications document. Also works as a place to resolve use case’s related conflicts and issues, discussing problems and challenges.
  + Document:
    - *#document-links:* for sharing documents’ links and storing for later uses.
    - *#meeting-notes:* acts as a communication method between the members in a meeting, as well as a place for the meeting’s secretary to upload a rough note of the result of the discussion for other members, similar to cloud-based, synchronized sticky notes.
* Voice channels for team meetings and pair programming.
* *Meeting:* for online, remote meeting sessions.
* *Study\_together:* for pair programming, as well as capable of holding a tutoring session.
  1. Jira



* Task and sprint management.
* Sprint planning and backlog management.
* Board Columns: TO DO, IN PROGRESS (DOCUMENT), IN PROGRESS (TECHNICAL), IN REVIEW, DONE
* Progress tracking through board movement. With this setup, we can have a clear view of what we need to do.
  1. Github



* The repository is integrated with Jira for automation.
* Main repository: BookLeaf
* Branch Strategy:
* main: stable production code
* [JIRA-ID]: Code or tasks from the Jira tasks.
* Folder Organization
* /src: Frontend code, Backend code, Database scripts, Configuration files
* /docs: used to store documentations
* /requirements: Project vision, User stories, System requirements
* /design: Architecture diagrams, UI/UX mockups, Database schemas
* /pa:
* /pa0, /pa1, /pa2
* [subsequent assignments]
  1. JIRA and GitHub Integration
* Task States and Transitions
* TODO → IN PROGRESS: When creating new branch
* IN PROGRESS → IN REVIEW: When creating pull request
* IN REVIEW → DONE: When merging pull request
* Automated Updates
* Branch creation updates JIRA status
* Pull requests link to JIRA tasks
* Merges automatically close tasks
* GitHub activities reflect in JIRA board