

THE UNIVERSITY OF BUEA
P.O BOX 63
Buea, Southwest Cameroon
Tel: (237) 674354327
Fax: (237) 3332 22 72



REPUBLIC OF CAMEROON
Peace -Work-FatherLand

FACULTY OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF COMPUTER ENGINEERING

**COURSE TITLE: SOFTWARE VERIFICATION TOOLS AND VALIDATION
TECHNIQUES**

COURSE CODE: CEF 474

25th April 2024

Verification and Validation Tools in Software Development Life Cycle: Figma and Git

Course Supervisor:

Dr. TSAGUE Aline

2023/2024

Group Members

	NAMES	MATRICULE NUMBERS
1.	ENOW MYKE-AUSTINE ETA	FE21A183
2.	FOLETIA AJONGAKUE SYNCLAIR	FE21A190
3.	INDAH RICOBELLE MBAH	FE21A204
4.	KENFACK SAMEZA	FE21A213
5.	MOKFEMBAM FABRICE KONGNYUY	FE21A240
6.	MONDOA ROBERT	FE21A241
7.	QUINUEL TABOT NDIP-AGBOR	FE21A300
8.	SIRRI THERESIA ANYE	FE21A306
9.	TAKEM JIM	FE21A309
10.	TSAPZE ZAMBOU ROSELINE	FE21A328

Contents

1. INTRODUCTION	4
2. FIGMA - DESIGN STAGE.....	4
2.1. Verification:.....	4
2.2. Validation:	7
3. GIT - DEVELOPMENT STAGE.....	7
3.1. Verification:.....	8
3.2. Validation:	9
4. CONCLUSION	10
5. REFERENCES	10

1. INTRODUCTION

In the software development life cycle (SDLC), verification and validation play crucial roles in ensuring the quality and correctness of software products. Verification involves checking whether the software meets specified requirements, while validation focuses on confirming that the software meets the user's needs and expectations. This report explores two powerful tools, Figma and Git, and their applications for verification and validation in the design and development stages of the SDLC.

2. FIGMA - DESIGN STAGE

Figma is a collaborative design tool that enables teams to create, share, and iterate on designs. It offers a range of features that support both verification and validation activities during the design stage.

Download Figma: (<https://www.figma.com/downloads/>)

2.1. Verification:

- **Real-time Collaboration:** Figma allows multiple designers to work simultaneously on a design, facilitating real-time collaboration. This feature enables designers to verify the consistency and coherence of design elements across the project.

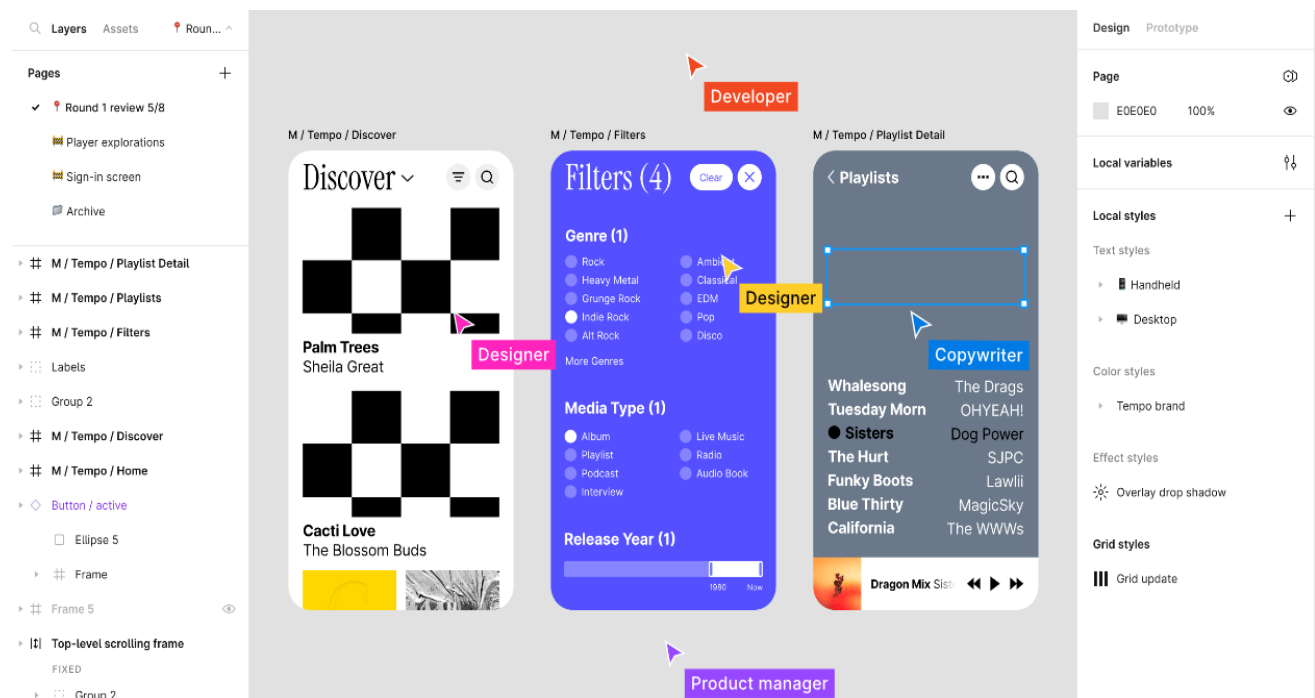


Fig 1: Figma Teams

- **Commenting and Feedback:** Figma provides a commenting system that allows designers and stakeholders to provide feedback directly on specific design components. This feature aids in verifying design decisions and gathering input from team members.

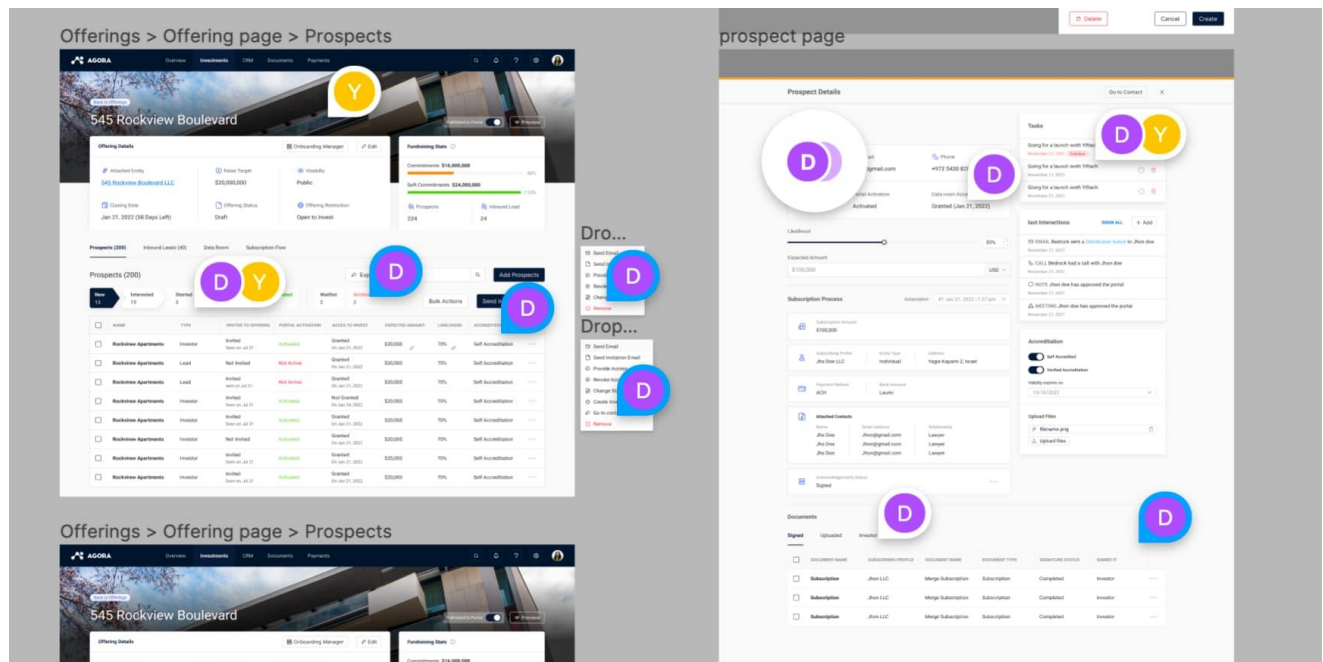


Fig 2: Figma comments

- **Version History:** Figma maintains an extensive version history, allowing designers to review and compare different iterations of a design. This capability enables verification by identifying changes, tracking progress, and reverting to previous versions if necessary.

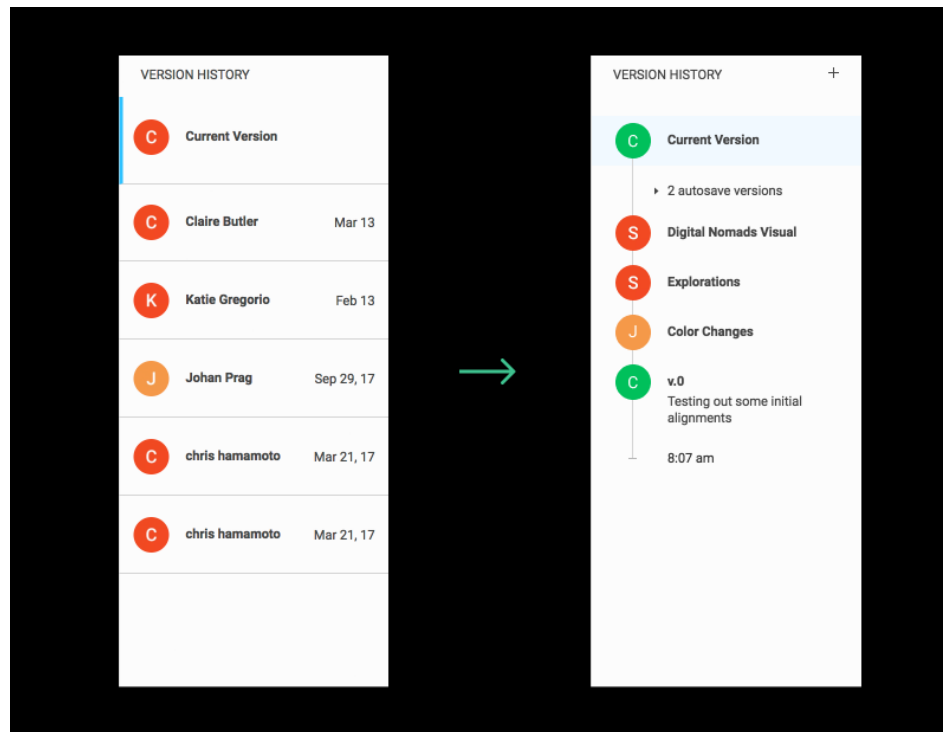


Fig 3: Figma Versions History

- **Design System Management:** Figma supports the creation and maintenance of design systems, which are collections of reusable design components and guidelines. By using design systems in Figma, teams can ensure consistency across designs and verify that the design follows established standards.

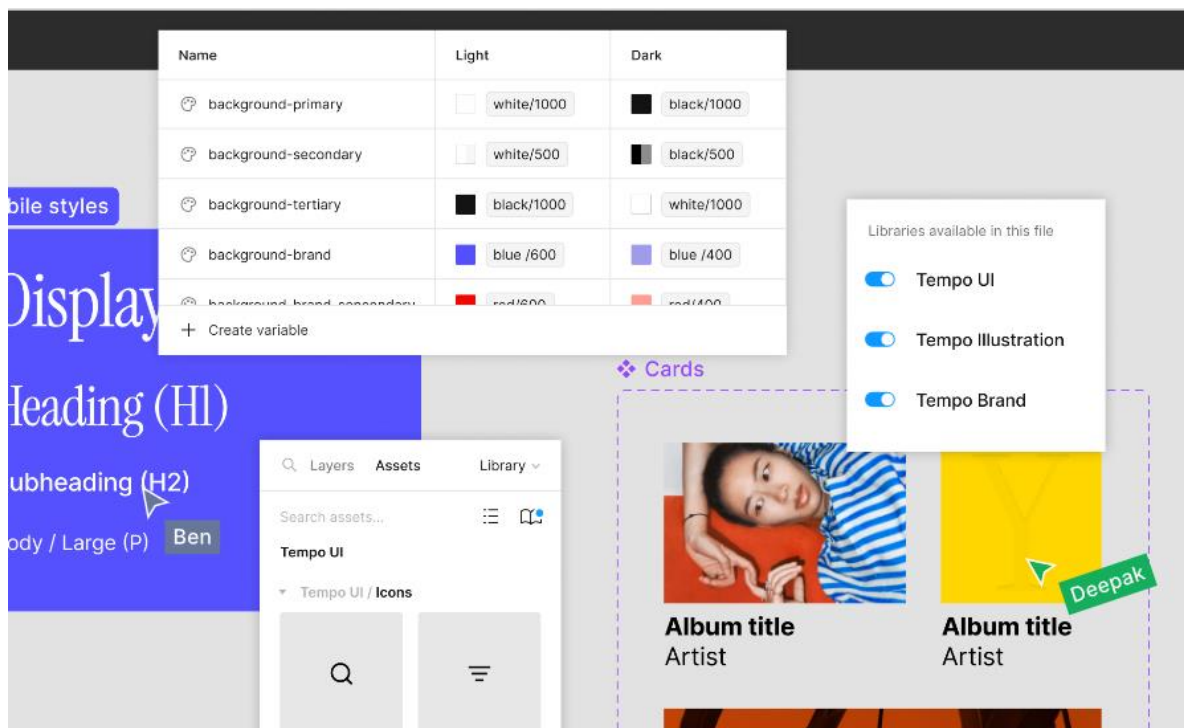


Fig 4: Figma Design System Management, with reusable colours and icons.

2.2. Validation:

- **Design Prototyping:** Figma supports the creation of interactive design prototypes. Designers can simulate user interactions and validate the user experience by testing the prototype with stakeholders or conducting usability studies.

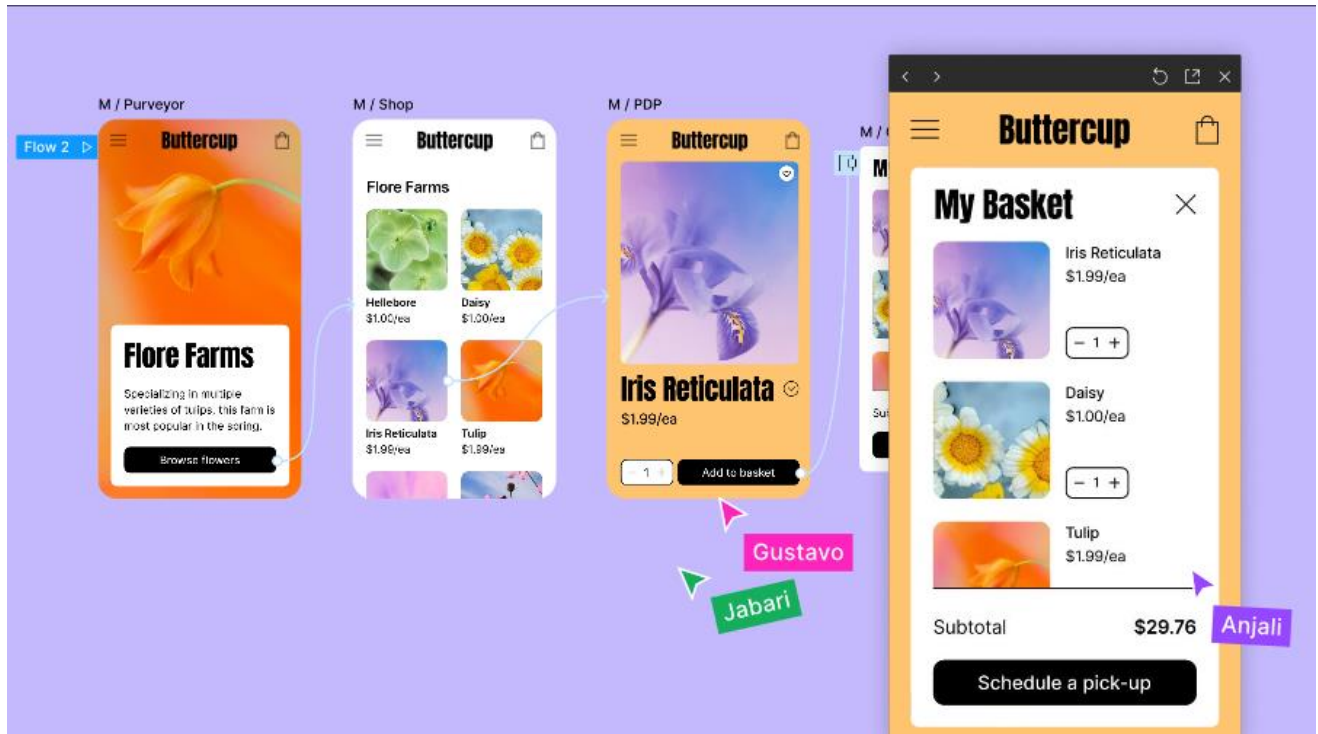


Fig 5: Figma Prototypes

- **User Testing:** Figma's prototyping capabilities enable designers to share interactive prototypes with users for feedback and validation. This facilitates usability testing and helps ensure that the design meets the user's expectations.
- **Feedback Collection:** Figma allows designers to collect feedback directly from users and stakeholders through the prototype. This feature aids in validating design decisions and incorporating user insights into the design process.
- **Accessibility Testing:** Figma offers plugins and features that assist in evaluating the accessibility of designs. Designers can validate the compliance of their designs with accessibility standards and guidelines. E.g. Figma "Color Contrast Checker" is used to check for contrast.

3. GIT - DEVELOPMENT STAGE

Git is a distributed version control system widely used in software development. It offers a range of features that support verification and validation activities during the development stage.

Download Git: <https://git-scm.com/downloads>

3.1. Verification:

- **Code Review:** Git enables peer code reviews by providing features like pull requests. Developers can review each other's code changes, identify errors, suggest improvements, and ensure code quality before merging it into the main codebase.

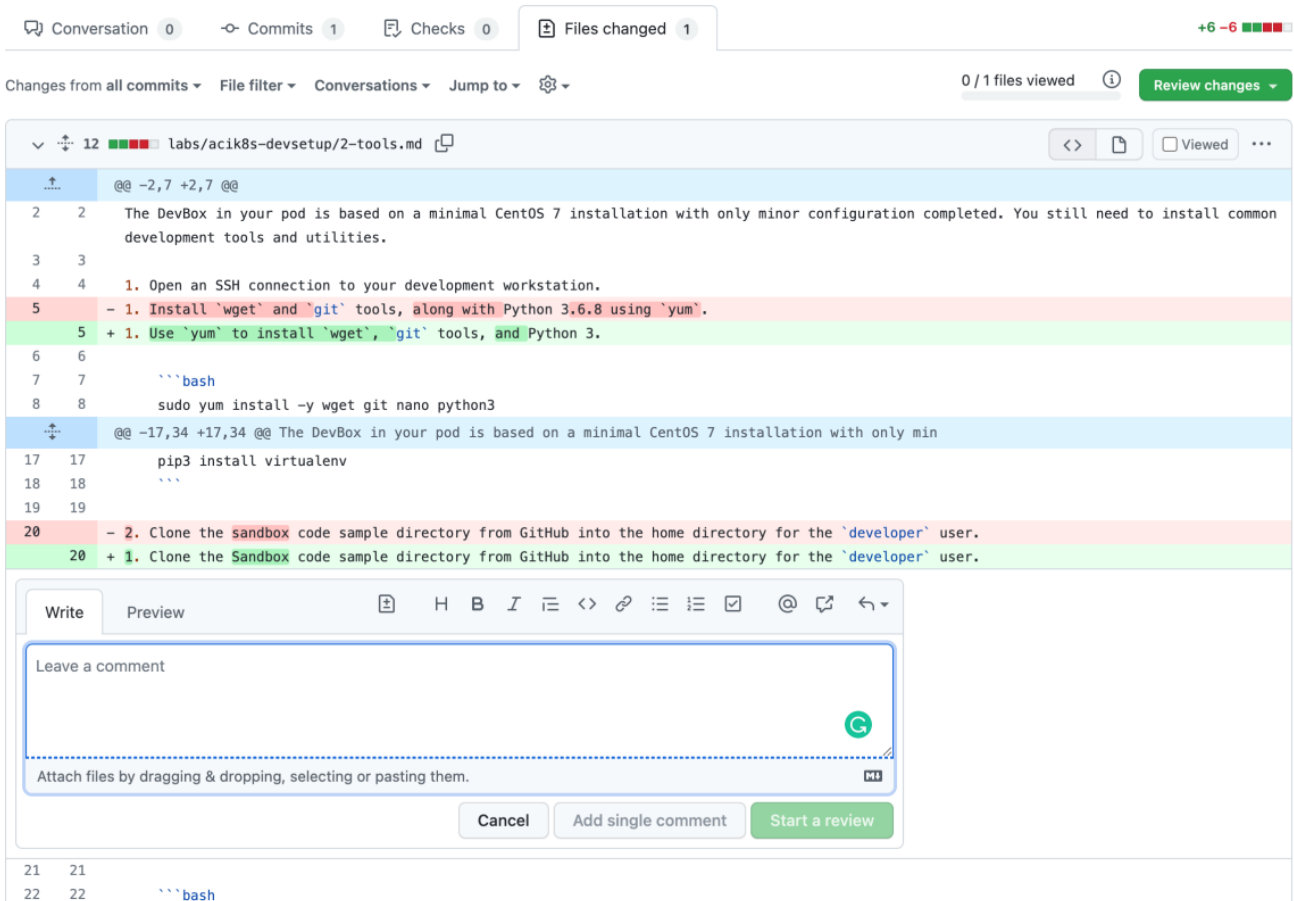


Fig 6: Git Code Review

- **Continuous Integration (CI):** Git integrates seamlessly with CI tools like Jenkins, Travis CI, or CircleCI. This integration enables automated verification processes such as building, testing, and analyzing code to identify bugs or issues early in the development cycle.
- **Branching Strategies:** Git's branching and merging capabilities allow teams to implement different branching strategies, such as feature branching or release branching. These strategies support the verification of code changes in isolated branches before merging them into the main codebase.

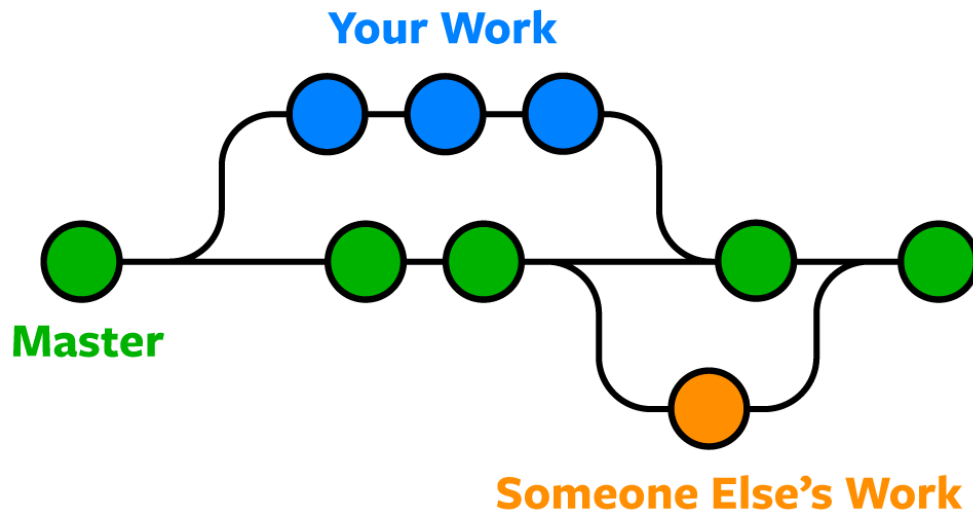


Fig 7: Git Branch Sample Structure

- **Version Control:** Git provides robust version control capabilities, allowing developers to track and manage changes to source code. This ensures that the software artifacts are correctly versioned and can be verified against specific requirements or bug fixes
- **Issue Tracking:** Git seamlessly integrates with issue tracking systems like Jira or GitHub Issues. This integration enables developers to link code changes to specific issues, track bug fixes, and ensure that all identified issues are addressed.
- **Documentation:** Git can also be used to store documentation related to the software requirements, design, and testing procedures. By maintaining documentation in the repository, developers can validate that the software aligns with the specified requirements and has been tested according to the defined criteria.
- **Collaboration:** Git enables seamless collaboration among developers through features like pull requests, code reviews, and issue tracking. These collaboration capabilities facilitate knowledge sharing, feedback incorporation, and validation of code changes by multiple team members.

3.2. Validation:

- **Documentation:** Git supports the management of project documentation alongside the source code. Developers can create and maintain documentation files, such as READMEs or user guides, within the repository. This helps validate the software by providing clear instructions and context to users and stakeholders.

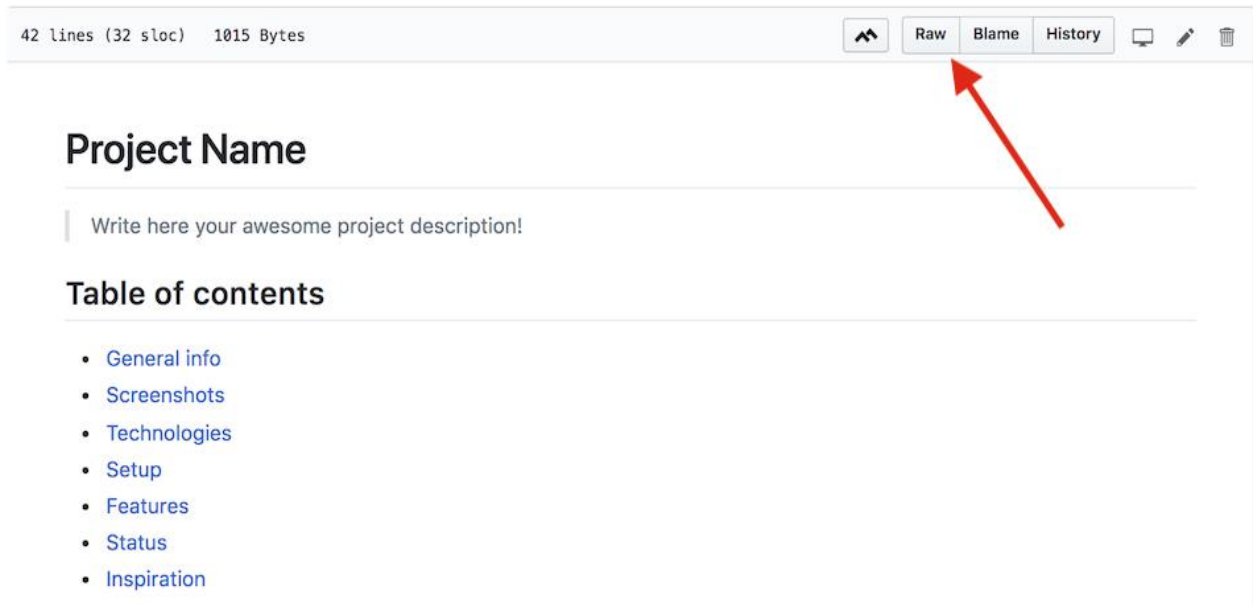


Fig 8: Git ReadMe file Sample

- **Code Reviews:** Code reviews in Git can serve as a form of validation by allowing team members to verify that the code changes align with the project requirements, coding standards, and best practices. Code reviewers can provide feedback, suggest improvements, and ensure that the code meets the desired functionality.
- **Collaboration:** Collaboration through Git facilitates validation by enabling team members to work together to review code changes, discuss requirements, and ensure that the software meets the desired specifications. By collaborating through Git, teams can validate the correctness and quality of the software collectively.

4. CONCLUSION

Figma and Git are powerful tools that significantly contribute to the verification and validation processes in different stages of the SDLC. Figma's design capabilities, including real-time collaboration, commenting, version history, and prototyping, support the verification and validation of design decisions. Git's version control, code review, CI, branching strategies, issue tracking, documentation, and collaboration features enhance the verification and validation of code changes during the development stage. By leveraging these tools effectively, software development teams can improve the quality and reliability of their products.

5. REFERENCES

- I. Figma: (<https://www.figma.com/>)
- II. Git: <https://git-scm.com/>