**Software Engineering Lab Experiment No. 4**

**Aim:** Create class diagram for the project

**Objectives:**

1. The objective of this lab experiment is to create a class diagram for the software project.
2. Providing a visual representation of the structure and relationships between classes.
3. This class diagram will serve as a reference for developers and other stakeholders to understand the architecture of the system.

**Requirements:**

1. Computer with internet access.
2. A Figma account.
3. Sample software project or problem statement for requirements analysis.
4. Word processing software for creating the lab report.

**Concept:**

1. **Class Diagram**

* A class diagram is a type of UML (Unified Modeling Language) diagram that provides a visual representation of the structure and relationships within a system or software application.

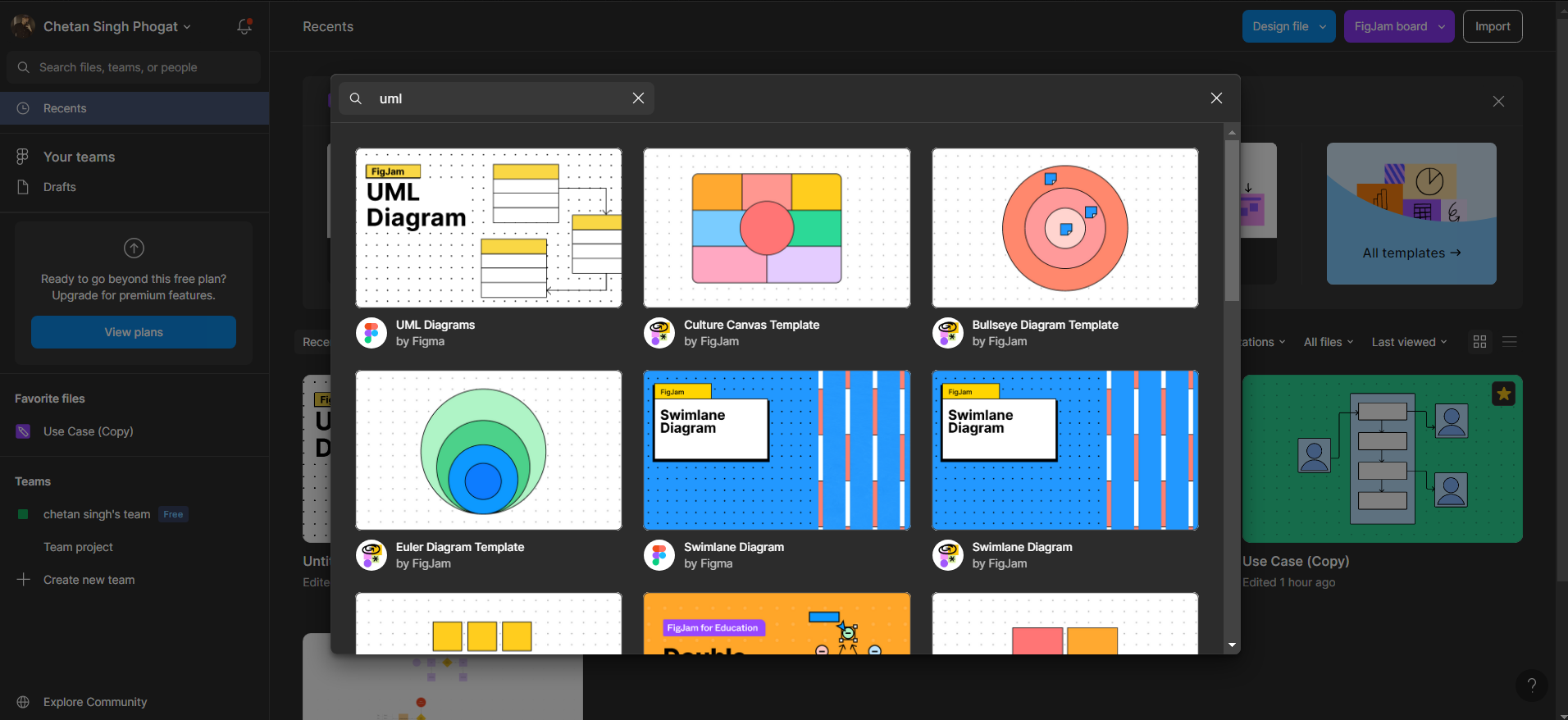
**Tools:**

* **Figma:** Figma is a web-based design tool that allows users to create interactive user interface prototypes. It enables real-time collaboration and can be accessed from anywhere and on any device.

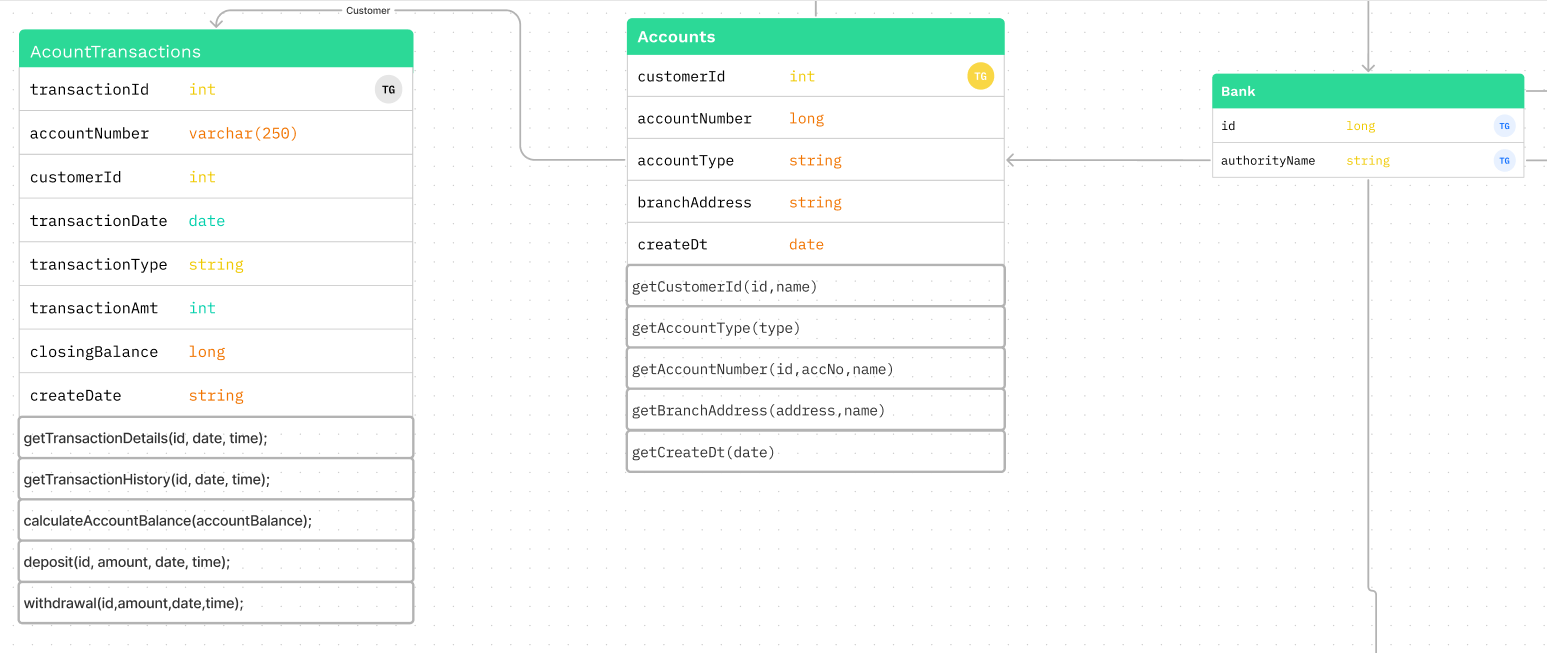
**Steps:** ***General steps for creating a Class diagram:***

1. **Identify Classes:** Begin by identifying the key classes in your project. These classes represent the major components, entities, and functionalities in your software system.
2. **Define Attributes and Methods:** These attributes and methods represent the state and behaviour of each class.
3. **Identify Relationships:** Determine the relationships between classes. Common relationships include associations, aggregations, compositions, and inheritances. Understand how classes are connected and interact with each other.
4. **Select UML Diagramming Tool:**

**Figma:** Figma is a web-based design tool that allows users to create interactive user interface prototypes.



1. **Create the Class Diagram:** Use a UML Modeling tool or drawing software to create the class diagram. Place classes, attributes, methods, and relationships on the diagram. Use appropriate symbols and notations to represent these elements.



1. **Review and Validate:**

Review the class diagram with the project team and other stakeholders. Ensure that it accurately reflects the project's design and that all relationships and dependencies are correctly represented.

**Conclusion:**

* The class diagram provides a clear visual representation of classes, their attributes, methods, and relationships. By following the outlined steps, we have successfully documented the system's design, allowing developers and stakeholders to have a common understanding of the project's structure.

**References:**

- Figma - Collaborative interface design tool (https://www.figma.com/)

- Software Engineering: A Practitioner's Approach by Roger S. Pressman (For further reading on requirements analysis in software engineering).