Lab report

Course title: OOAD Lab
Course code: CSE-314
3 rd Year 1st Semester 2022

Date of Submission: 17/08/2023

Submitted to-

Sarnali Basak
Associate Professor

Nadia Afrin Ritu
Lecturer

Department of Computer Science and Engineering
Jahangirnagar University
Savar, Dhaka-1342

Class roll	Exam roll	Name
346	202158	Solaimi Hamid
351	202263	Umma Sumaiya Jahan
371	202183	Mamunur Roshid

400	202212	Mohammad Rokibul Hassan Shanto
404	202216	Md. Mahfuzur Rahman

Feedbacks based on experience:

1. Lucidchart:
    - **Features**: Real-time collaboration, extensive template library, integration with Google Workspace and Microsoft Office, UML diagram support.
    - **Strong Points:** User-friendly interface, collaborative capabilities, integration with popular productivity tools.
    - **Weak Points**: Some advanced features are behind a paywall, might be overwhelming for simpler diagramming needs.
    - **Comparison:** Better for collaboration and integration compared to Draw.io, but might be pricier for advanced features.

2. Draw.io:
    - **Features:** Open-source, supports various diagram types, offline usage, multiple export options.
    - **Strong Points**: Free and open-source, versatile diagramming capabilities.
    - **Weak Points:** User interface might not be as polished, lacks some advanced features.
    - **Comparison:** Great for open-source enthusiasts, but might lack the refinement of Lucidchart in terms of interface and collaboration.

3. Visual Paradigm:
    - **Features:** Supports multiple modeling languages, integration with development processes, version control, code generation.
    - **Strong Points:** Comprehensive modeling capabilities, ideal for complex projects and enterprise teams.
    - **Weak Points:** Steeper learning curve, may be too advanced for smaller projects.
    - **Comparison:** Offers more advanced modeling capabilities compared to Lucidchart and Draw.io, but might not be as user-friendly for beginners.

4. Enterprise Architect:
    - **Features:** Extensive modeling language support, enterprise-level collaboration and integration.
    - **Strong Points:** Designed for complex projects and large teams, comprehensive modeling capabilities.

- **Weak Points:** High learning curve, expensive pricing, might be overwhelming for smaller projects.
- **Comparison:** More suitable for large-scale projects than most other tools due to its advanced features and enterprise-oriented focus.

5. Creately:
    - **Features:** Real-time collaboration, various diagram templates, UML diagram support.
    - **Strong Points:** User-friendly, great for smaller teams and rapid diagram creation.
    - **Weak Points:** Limited advanced features compared to some competitors.
    - **Comparison:** Offers collaborative features similar to Lucidchart but may lack the advanced capabilities of more specialized tools.

6. PlantUML:
    - **Features:** Text-based diagram creation, code-based integration, open-source.
    - **Strong Points:** Excellent for developers wanting to generate diagrams from code, free and version-control-friendly.
    - **Weak Points:** Text-based interface might be less intuitive for non-developers.
    - **Comparison:** Uniquely suited for code-based diagram generation, but not as visually intuitive as graphical tools.

7. Microsoft Visio:
    - **Features:** Integration with Microsoft Office, UML diagram support, user-friendly.
    - **Strong Points:** Familiar interface for Microsoft users, seamless integration with Office suite.
    - **Weak Points:** Costly, might lack some advanced features of specialized tools.
    - **Comparison:** Better for users within the Microsoft ecosystem, but might not be as feature-rich as some alternatives.

8. Gliffy:
    - **Features:** Real-time collaboration, UML diagram support, integration with Confluence and Jira.
    - **Strong Points:** Simplicity, easy collaboration, integration with Atlassian products.
    - **Weak Points:** Might lack advanced features and extensive capabilities.
    - **Comparison:** Offers better integration with Atlassian products compared to Lucidchart, but may not be as versatile.

9. SmartDraw:
    - **Features:** Automation features, various diagram templates, UML diagram support.

- **Strong Points:** Quick diagram creation, automation capabilities, professional-looking diagrams.
- **Weak Points:** Pricing might be higher, advanced features might not be as robust.
- **Comparison:** Suitable for rapid creation of professional diagrams, but might not match the comprehensive capabilities of Visual Paradigm.

10. yEd Graph Editor:
    - **Features:** Automatic layout algorithms, various diagram types.
    - **Strong Points:** Open-source, automatic layout for clean diagrams.
    - **Weak Points:** Limited collaboration features, might have a less modern interface.
    - **Comparison:** Better for automatic layout and open-source enthusiasts, but may lack the collaboration features of cloud-based tools.

Comparison of these diagramming tools based on various factors:

1. Collaboration and Teamwork:
    - Lucidchart, Creately, Gliffy, and Visual Paradigm offer strong real-time collaboration features, making them suitable for teams working together on diagrams.
    - Draw.io (diagrams.net) and yEd Graph Editor might have slightly limited collaborative features compared to some other cloud-based tools.
    - Enterprise Architect is well-equipped for large teams and enterprise-level collaboration.

2. Ease of Use:
    - Lucidchart, Creately, and Microsoft Visio are known for their user-friendly interfaces, suitable for users of varying skill levels.
    - Draw.io (diagrams.net) and yEd Graph Editor might have slightly steeper learning curves due to their interfaces.
    - PlantUML requires familiarity with its text-based syntax, which might be a hurdle for non-developers.

3. Diagram Variety and Complexity:
    - Visual Paradigm, Enterprise Architect, and SmartDraw excel in handling complex diagrams and supporting a wide range of modeling languages.

- Draw.io (diagrams.net), Gliffy, and yEd Graph Editor are suitable for simpler diagramming needs and might lack some advanced capabilities.
- PlantUML focuses on creating diagrams from code and might not be as versatile for complex diagrams.

4. Integration and Compatibility:
    - Lucidchart, Gliffy, and Microsoft Visio integrate well with other popular productivity tools like Microsoft Office and Atlassian products.
    - Visual Paradigm and Enterprise Architect offer robust integration with development lifecycles and tools.
    - Draw.io (diagrams.net) and PlantUML might offer fewer integrations compared to some commercial tools.

5. Cost and Pricing:
    - Draw.io (diagrams.net), PlantUML, and yEd Graph Editor are open-source or free, making them cost-effective choices.
    - Lucidchart, Visual Paradigm, and Enterprise Architect can be more expensive, especially for advanced features and larger teams.

6. Specialization:
    - Enterprise Architect is designed for enterprise-level projects and complex modeling needs.
    - PlantUML is uniquely suited for developers who want to generate diagrams from code.
    - SmartDraw is tailored for quickly creating professional-looking diagrams.

7. Platform Preference:
    - Microsoft Visio is preferable for users within the Microsoft ecosystem due to its seamless integration.
    - Draw.io (diagrams.net) and yEd Graph Editor are desktop applications, while others are primarily cloud-based.

8. Open-Source and Customization:
    - Draw.io (diagrams.net), PlantUML, and yEd Graph Editor are open-source and allow for customization.
    - Other tools provide customizable features, but open-source tools offer greater flexibility for developers.

In summary, the choice among these tools depends on your specific requirements. For collaborative projects, Lucidchart, Visual Paradigm, and Enterprise Architect are strong contenders. For open-source enthusiasts, Draw.io (diagrams.net), PlantUML, and yEd Graph Editor offer customization and flexibility. If ease of use and compatibility with Microsoft Office

are crucial, consider Microsoft Visio. Always evaluate based on factors like project complexity, team size, collaboration needs, budget, and familiarity with the tool's interface.