

Practical File Of Software Management Tools

**BACHELOR OF TECHNOLOGY
COMPUTER SCIENCE AND ENGINEERING**



**Submitted By :
Aman Chauhan
1805158**

Department of Computer Science and Engineering
Guru Nanak Dev Engineering College

Table Of Contents

Sr. No.	Practical	Page No.
1.	Introduction to software management tools	3
2.	Introduction and familiarization with Github	5
3.	Familiarization with smart draw software with examples	7
4.	Familiarization with UML with examples	9
5.	Write in detail about reference manager tools, taking mendeley as an example	13
6.	Introduction To Bugzilla	15
7.	Introduction To Jenkins	18
8.	Introduction To Harvest	21
9.	Introduction To Winrunner	23

Practical 1

Introduction to software management tools.

Introduction:

Software management is a discipline of planning and supervising software projects. It is a sub-discipline of software project management in which software projects are planned, implemented, monitored and controlled. It is a procedure of managing, allocating and timing resources to develop computer software that fulfills requirements.

There are three needs for software project management. These are:

- Time
- Cost
- Quality

It is an essential part of the software organization to deliver a quality product, keeping the cost within the client's budget and deliver the project as per schedule. There are various factors, both external and internal, which may impact this triple factor. Any of the three-factors can severely affect the other two.

Software Management Activities:

Software management comprises a number of activities, which contains planning of project, deciding scope of software product, estimation of cost in various terms, scheduling of tasks and events, and resource management. Project management activities may include:

- Project Planning
- Scope Management
- Project Estimation

Benefits of Software Management Tools:

- Project Planning
- Improved Productivity
- Effective Assignment of Tasks
- Better team collaboration
- Budget Management
- Progress Monitoring
- Document Sharing and Access

Examples:

- GitHub
- GitLab
- Bitbucket
- GitBucket
- AWS Code Commit

Practical 2

Introduction and familiarization with Github.

Github

GitHub is an immense platform for code hosting. It supports version controlling and collaboration and allows developers to work together on projects. It offers both distributed version control and source code management (SCM) functionality of Git. It also facilitates collaboration features such as bug tracking, feature requests, task management for every project.

Essential components of the GitHub are:

- Repositories
- Branches
- Commits
- Pull Requests
- Git (the version control tool GitHub is built on)

Advantages of GitHub

- The key benefits of GitHub are as follows.
- It is easy to contribute to open source projects via GitHub.
- You can attract the recruiter by showing off your work. If you have a profile on GitHub, you will have a higher chance of being recruited.
- It allows your work to get out there in front of the public.
- You can track changes in your code across versions.

Features of GitHub

- Collaboration
- Integrated issue and bug tracking
- Graphical representation of branches
- Git repositories hosting
- Project management
- Team management
- Code hosting
- Track and assign tasks
- Conversations

Practical 3

Familiarization with smart draw software with examples.

SmartDraw is a web-based diagramming tool used by teams to collaborate on and make flowcharts, organization charts, mind maps, project charts, and other business visuals.

SmartDraw includes quick-start templates for over 70 different diagram types from flowcharts to floor plans. Powerful automatic formatting means perfect layouts in minutes. It's the right fit whether you're working on your own or collaborating with a large enterprise team.

Build diagrams like org charts, ERD, and class diagrams from data with built-in extensions or enhance any diagram with shape data and generate manifests. You can also connect to your AWS account and generate a visual automatically.

Integration and Add-Ons:

- SmartDraw lets you add diagrams to Microsoft Office products including Word, PowerPoint, and Excel and Google Workspace applications like Google Docs and Google Sheets.
- SmartDraw has apps for Atlassian's Confluence, Jira, and Trello.
- SmartDraw lets users save files to Sharepoint, OneDrive, Google Drive, Dropbox, and Box.
- SmartDraw also integrates with Microsoft Teams and can generate diagrams using data from Azure and AWS.

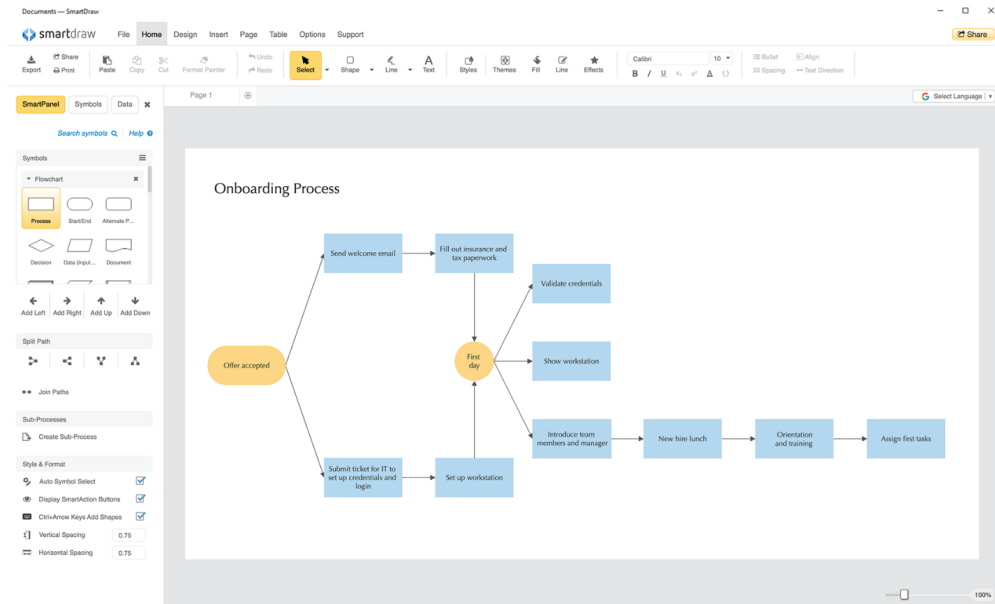


Fig : Smart Draw

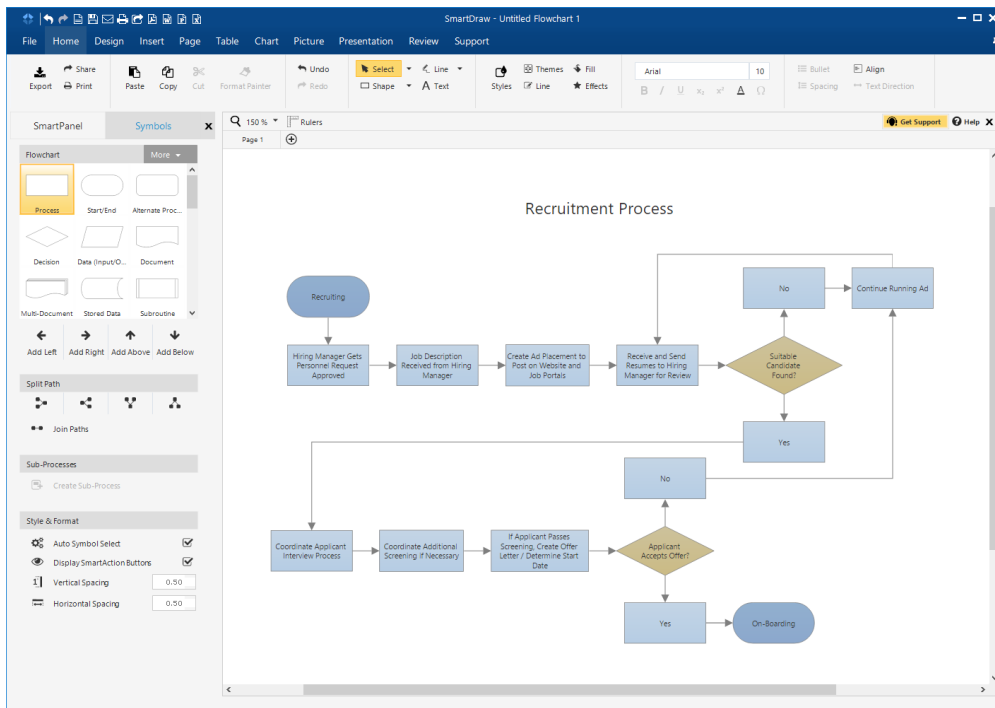


Fig : Smart Draw

Practical 4

Familiarization with UML with examples.

Introduction Of UML:

The UML stands for Unified modeling language, is a standardized general-purpose visual modeling language in the field of Software Engineering. It is used for specifying, visualizing, constructing, and documenting the primary artifacts of the software system. It helps in designing and characterizing, especially those software systems that incorporate the concept of Object orientation. It describes the working of both the software and hardware systems.

Goals of UML:

- Since it is a general-purpose modeling language, it can be utilized by all the modelers.
- UML came into existence after the introduction of object-oriented concepts to systemize and consolidate the object-oriented development, due to the absence of standard methods at that time.
- The UML diagrams are made for business users, developers, ordinary people, or anyone who is looking forward to understanding the system, such that the system can be software or non-software.
- Thus it can be concluded that the UML is a simple modeling approach that is used to model all the practical systems.

Characteristics of UML:

- It is a generalized modeling language.
- It is distinct from other programming languages like C++, Python, etc.
- It is interrelated to object-oriented analysis and design.
- It is used to visualize the workflow of the system.
- It is a pictorial language, used to generate powerful modeling artifacts.

UML-Diagrams:

The UML diagrams are categorized into structural diagrams, behavioral diagrams, and also interaction overview diagrams. The diagrams are hierarchically classified in the following figure:

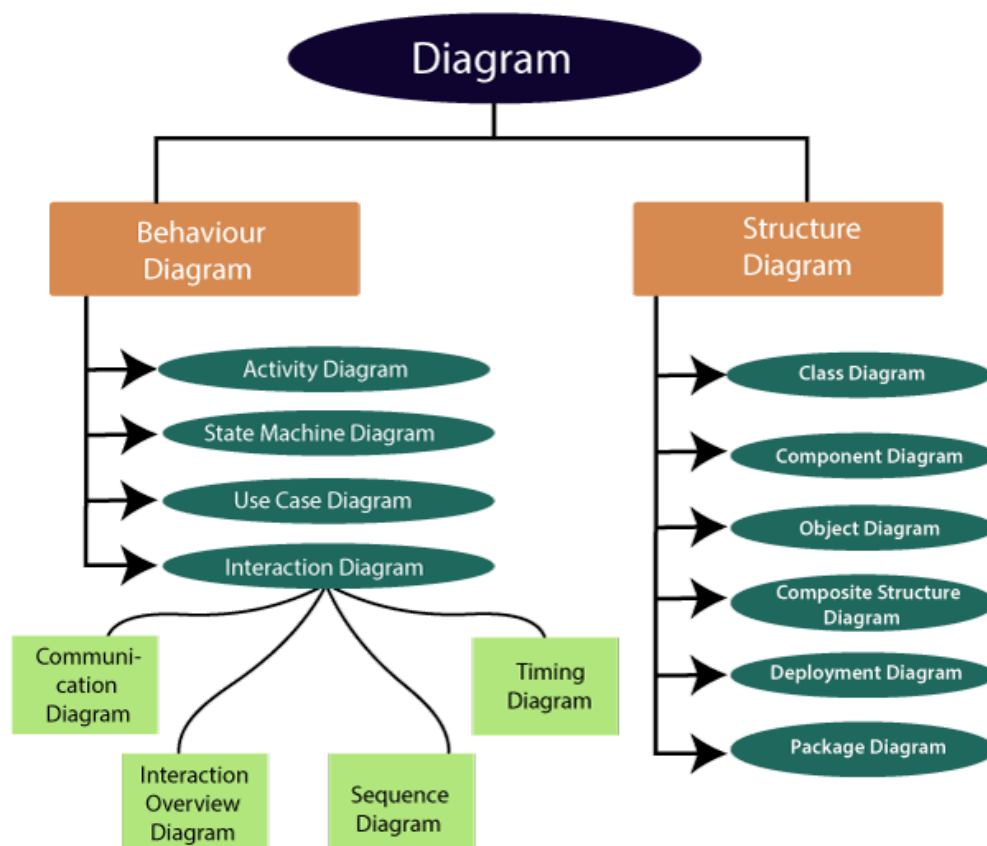


Fig: Diagram

1. Structural Diagrams:

Structural diagrams depict a static view or structure of a system. It is widely used in the documentation of software architecture. It embraces class diagrams, composite structure diagrams, component diagrams, deployment diagrams, object diagrams, and package diagrams.

- **Class Diagram:** Class diagrams are one of the most widely used diagrams. It is the backbone of all the object-oriented software systems. It depicts the static structure of the system.
- **Composite Structure Diagram:** The composite structure diagrams show parts within the class. It displays the relationship between the parts and their configuration that ascertain the behavior of the class.
- **Object Diagram:** It describes the static structure of a system at a particular point in time. It can be used to test the accuracy of class diagrams.
- **Component Diagram:** It portrays the organization of the physical components within the system. It is used for modeling execution details.
- **Deployment Diagram:** It presents the system's software and its hardware by telling what the existing physical components are and what software components are running on them.
- **Package Diagram:** It is used to illustrate how the packages and their elements are organized. It shows the dependencies between distinct packages.

2. Behavioral Diagrams:

Behavioral diagrams portray a dynamic view of a system or the behavior of a system, which describes the functioning of the system. It includes use case diagrams, state diagrams, and activity diagrams.

- **State Machine Diagram:** It is a behavioral diagram. it portrays the system's behavior utilizing finite state transitions. It is also known as the **State-charts** diagram.
- **Activity Diagram:** It models the flow of control from one activity to the other. With the help of an activity diagram, we can model sequential and concurrent activities.
- **Use Case Diagram:** It represents the functionality of a system by utilizing actors and use cases. It encapsulates the functional requirement of a system and its association with actors.

3. Interaction Diagrams

It shows how objects interact with each other and how the data flows within them. It consists of communication, interaction overview, sequence, and timing diagrams.

- **Sequence Diagram:** It shows the interactions between the objects in terms of messages exchanged over time.
- **Communication Diagram:** It shows the interchange of sequence messages between the objects. It focuses on objects and their relations.
- **Timing Diagram:** It is a special kind of sequence diagram used to depict the object's behavior over a specific period of time.
- **Interaction Overview diagram:** It is a mixture of activity and sequence diagram that depicts a sequence of actions to simplify the complex interactions into simple interactions.

Practical 5

Write in detail about reference manager tools, taking mendeley as an example.

For a research paper to get published, it must have novel data. Therefore, when a researcher designs experiments and writes manuscripts, information and use of previous studies are fundamental. Indeed, a scientist must know what is known before designing an experiment. To help manage all of the research articles and studies that scientists sift through daily, there are citation managers that help organize papers.

Originally, reference managers helped researchers correctly format their citations. These tools have come a long way since then.

Reference management programs (also known as citation managers or bibliographic management software) provide many useful features that can help you with your research such as:

- storing and organizing your references
- generating citations and bibliographies in the style you prefer; and,
- easily converting referencing styles to suit publication requirements.

Mendeley is a free reference manager that can help you collect references, organize your citations, and create bibliographies. The strength of Mendeley, however, is what adds to that. Mendeley is also an academic social network that enables you to share your research with others.

The free edition has up to 2GB of personal storage.

With Mendeley Reference Manager you can:

- Store, organize and search all your references from just one library.
- Seamlessly insert references and bibliographies into your Microsoft® Word documents using Mendeley Cite.
- Read, highlight and annotate PDFs, and keep all your thoughts across multiple documents in one place.
- Collaborate with others by sharing references and ideas.

Practical 6

Introduction To Bugzilla.

Introduction:

Bugzilla is a bug tracking tool that helps to track the issues related to their product.

Bugzilla tool is written in Perl language, and it uses MySQL database.

It is a bug tracking tool. However, it can also be used as a test management tool because it can be linked with other Test case management tools such as Quality Center, Testlink, etc. It is an open source tool, i.e., this tool is available to the users at a free of cost.

Features of Bugzilla:



Fig : Features

- **Advanced Search capabilities**

Bugzilla has two forms of search:

- It has Google-like bug search which is simple to use, and it also searches the full text of a bug.
- It provides you with a very advanced search system where you can create any type of search that you want such as time-based searches (For example, you want to see the list of bugs whose priority has been changed since the last two days).

- **Email Notifications controlled by user preferences**

You will get an email if any changes are made in the Bugzilla, and the notifications that you get on which bug is totally controlled by the user preferences.

- **Bug lists in multiple formats**

Reports and Charts When you search for the bugs, then you can get the bug lists in multiple formats such as Atom, iCalendar format.

- **Scheduled reports by email**

Bugzilla has a system that will send you, users or a group that you specify the results of a search on a schedule that you have mentioned.

- **Automatic Duplicate Bug Detection**

When you are filing a new bug, and when you type the summary for the bug, then the system looks for similar bugs. If the system finds the similar bugs,

then it allows the user to add themselves in the CC list of one of those bugs instead of creating a new one.

- **File/Modify Bugs by email**

You can send an email to Bugzilla to create a new bug or modify the existing bug. You can even attach the files to Bug.

- **Time Tracking**

Bugzilla also provides the feature of time tracking. You can determine how many hours a bug will take to get fixed and you can also even track the hours that you need to spend on the bug.

- **Request System**

The request system is a way of asking other users to do something with a bug. The user can either grant the request or deny the request, and Bugzilla will keep track of the answer.

- **Move Bugs Between Installations**

Bugzilla has the ability to move the bug from one Bugzilla installation to another. In Bugzilla, a bug can also be moved across different versions.

- **"Watch" Other Users**

Bugzilla allows you to watch other users. You will also get an email that the user gets from Bugzilla.

Practical 7

Introduction To Jenkins.

Introduction:

Jenkins builds and tests our software projects which continuously makes it easier for developers to integrate changes to the project, and makes it easier for users to obtain a fresh build.

It also allows us to continuously deliver our software by integrating with a large number of testing and deployment technologies.

With the help of Jenkins, organizations can speed up the software development process through automation. Jenkins adds development life-cycle processes of all kinds, including build, document, test, package, stage, deploy static analysis and much more.

For example: If any organization is developing a project, then Jenkins will continuously test your project builds and show you the errors in early stages of your development.

Possible steps executed by Jenkins are for example:

- Perform a software build using a build system like Gradle or Maven Apache
- Execute a shell script
- Archive a build result
- Running software tests

Workflow:

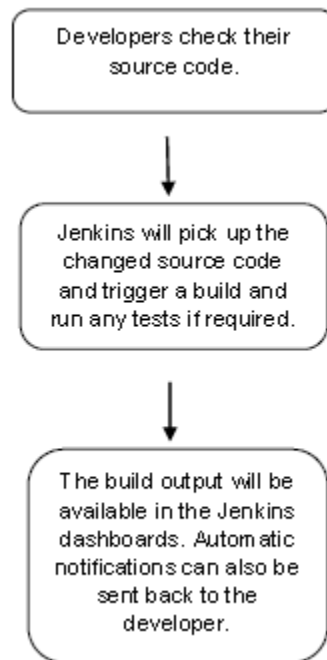


Fig : Workflow

Advantages of Jenkins:

- It supports 1000 or more plugins to ease your work. If a plugin does not exist, you can write the script for it and share with the community.
- It is platform independent. It is available for all platforms and different operating systems. Like OS X, Windows or Linux.
- Easy support, since it is open source and widely used.
- Jenkins also supports cloud based architecture so that we can deploy Jenkins in cloud based platforms.

Disadvantages of Jenkins:

- Its interface is outdated and not user friendly compared to current user interface trends.
- Not easy to maintain because it runs on a server and requires some skills as server administrator to monitor its activity.
- CI regularly breaks due to some small setting changes. CI will be paused and therefore requires some developer's team attention.

Practical 8

Introduction To Harvest.

Harvest is a web-based time tracking tool developed and launched by Iridesco LLC in 2006.

Harvest offers time tracking, invoicing, expense tracking, and time-based reporting. Users can send automated payment reminders from the software if clients haven't paid an invoice on time. This is a "less stressful option for managers who hate dunning their customers."

Harvest was one of the first software as a service (SaaS) applications to be built on the Ruby on Rails framework, and is listed as one of the most prolific by its creators. It was also one of the first businesses to integrate with Twitter, enabling its users to track time via tweets

- **Track time from browser, desktop, and mobile:**

Make it as easy as possible for your team to capture their time with dedicated apps across devices.

- **Integrated with your workflow:**

Harvest integrates with the tools your team knows and loves — capture time the way your team already works.

- **Custom reminders for consistent time tracking:**

Create automated reminders to help your team track time regularly and accurately.

- **Keep budgets on target:**

Harvest updates budgets as your team tracks time, so you can keep your projects on track and profitable.

- **Visualize team capacity:**

See who's overworked and who's under-utilized at a glance with capacity reporting.

- **Understand costs:**

Keep track of internal costs and review past project data to inform future project scope and estimates.

- **Analyze time spent:**

See which projects and tasks are consuming your team's time and impacting costs.

- **Dive into the details:**

Powerful filters give you the detailed data needed to understand where your time goes and plan what's next.

- **Do more with your data:**

It's easy to export data from Harvest into the reporting tools you already use to gain additional insights.

- **Streamline your invoicing:**

Easily create and send accurate invoices using your team's time and expenses already tracked in Harvest.

- **Get paid fast with online payments:**

PayPal and Stripe integrations let your clients quickly and conveniently pay invoices online.

Practical 9

Introduction To Winrunner.

WinRunner is an Automation Software Testing Tool that is owned by HP and was developed by Mercury Interactive. It is known and extensively used for its ability to support the majority of the programming languages and web development technologies such as C, C++, C#, Visual Basic, VC++, D2K, Java, HTML, PowerBuilder, Delphi, Cibell, etc. It is used for performing various testing techniques, which include functional testing, user interface testing, integration testing, regression testing, etc., by making use of its options to record the functionality for creating test steps and by using the playback UI interactions option for generating the test scripts.

WinRunner is a Functional Testing tool to work on a compilation of tests that are used for collaborating with the HP QuickTest Professional and to be used as a supporting element for the quality assurance process, as a part of the Test phase in the Software Development Life Cycle. This testing software is applied as a part of progress in product excellence.

The testing process here goes by capturing the functional requirement/ test requirement, by verifying the actual results against the expected results and by replaying the user operations/ functional activities performed on the software product. In this process, the tool can involuntarily perform the complete testing process while identifying the defects with respect to the product design provided by the business/ client personnel. The WinRunner Automation tool uses the Test Script Language (TSL) that is identical to the C programming language in terms of receiving the user activities as the process input and by allowing remarkable flexibility for modification and applying a constraint to the same.

The recording process is an insightful method that facilitates the generation of vigorous functional test items. In order to create the same, the tool typically records the functional flow of the software application by imitating the user actions

throughout the recording process. It also allows the testing professional to straight-up revise& update the scripts produced to match the most functionality described in the functional requirement specification documentation.

The process proceeds by allowing the testers to add checkpoints in order to enable the comparison process for testing between the functionality assigned as the expected results with that of the functionality assigned as the actual results. These checkpoints can include the test criteria, the user interface features, images/ logo and the navigation flow in the form of URLs.

Advantages:

- Enables Rapid Testing
- Provides Consistency
- Reusability of tests
- Customizable for future changes

Disadvantages:

- Doesn't apply for Stress or Load or Scalability Testing.
- Doesn't support .net programming
- Tester should have programming knowledge/experience
- It doesn't support multimedia systems.