1 .Istallation of STAR UML

Step1:staruml can e download from any browser open official page select the OS version suitable for your system and click on download file.
Step2:Run exe.file
Setp3:After installation the star uml icon shows in desktop.
Step4:The software will open directly and it will ask for buy now option Or evaluation the options
Step5:The screen appears like this for the evaluated version
Step6:If you check the c driven you can see the star uml folder in the program files location
Step7:Open and create a uml diagram.

2)Fast Food Billing

1)Functional Requirements

FR1. Super Admin

Manage: Super Admin manage User Application, bills and customers.

FR2. System User

Manage: System user manage payment mode, transactions and receipts.

FR3. Customer

Customer checks the bill, make payments and print receipts.

FR4. Accountant

Accountant will generate the bills, request for payments and collect the payments.

Non Functional Requirements

- NFR1.Performance Requirements:
    - This system should remain accessible 24x7
    - At least 50 users should be able to access the system altogether at any given time
- NFR2. Security Requirements:
    - This system should be accessible only within the institute LAN
    - The database should not store any password in plain text - a hashed value has to be stored
- NFR3. Software Quality Attributes
- NFR4. Database Requirements

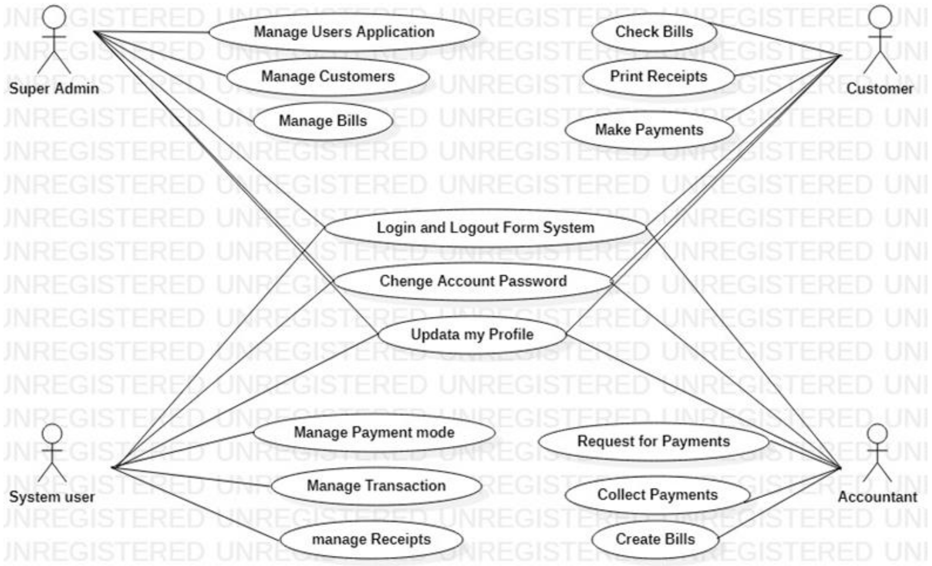
2. Risk

- **Schedule Risk**-Project schedules get slipped when project tasks and schedule release risks are not addressed properly.
- **Operational Risks**- Risk of loss due to improper process implementation, failed system or some external event risks.
- **Technical Risks**- Technical risks generally lead to failure of functionality and performance.

3. Use Case Story

- *As a Customer, I want online payments So that I can pay the bill fast.
- *As a customer I want hardcopy of bill, So that I can check whether it is correct or not.
- As a manager, I want a tablet so that I can check details on a booking quickly.
- As admin, I want customer contact information and even preferences, so that you can deliver a more personalized experience.
- As a staff, I want laptop so that it reduces manual work.

UML Diagram



4). Test suite:-
Test Suite 1:-Super Admin
Test Suite 1.1: Manage users
Test Suite 1.2: Manage Bills
Test Suite 1.3: Manage Customers

Test Suite 2:-Customer
Test Suite 2.1: Check Bills
Test Suite 2.2: print Receipts
Test Suite 2.3: Make payments

Test Suite 3:- System user
Test Suite 3.1: Manage Payment Mode
Test Suite 3.2: Transaction
Test Suite 3.3: Receipts

Test Suite 4:- Accountant
Test Suite 4.1: create Bills
Test Suite 4.2: Request for payments
Test Suite 4.3: collect Payments

3) HOTEL RESERVATION SYSTEM

For the past two years, Adams Beach Resort has been using manual billing and reservation system. Though they have an existing website, they do not update this because it is only for advertisement. Reservations in this website can be made by sending an e-mail to the manager of the resort, but because of the busy schedule of the manager, they cannot always monitor their mail and confirm reservations of clients. As a result, difficulties are experienced by the resort.

Adams Beach Resort is having a problem on Data Redundancy. Reservations through phone calls, e-mails and walk-ins are written repeatedly in their logbook. Cancellations of reservations are also written repeatedly in their logbook.

1.**Functional Requirements:**

1 Registration

FR1. The Customer should be able to register with their details

FR2. The system should record following customer details into member database.

FR3. The system shall send verification message to email

2 Logging In

FR4. The system should verify the customer email & password against the member database when logging in

FR5. After login, member should be directed to Home screen

3 Reservation

FR6. The system should enable customer to check for availability of rooms

FR7. The system should display rate for all rooms

FR8. The system should allow customer to confirm or cancel the booking

FR9. The system should record booking details into database

4 Receptionist Access

FR10. The system should allow Receptionist to update, add or delete booking information

FR11.The system should provide customer desk portal access to receptionist for providing response to customer inquiry

5 Manager Access

FR12. The system should generate financial and customer report for manager

FR13. The system should enable manager full modification access to customer, booking and room information

6 Payment Management Systems

FR14. The system should allow customer to pay bill via online using credit or debit card

Non Functional Requirements

- NFR1.Performance Requirements:
  - This system should remain accessible 24x7
  - At least 50 users should be able to access the system altogether at any given time
- NFR2. Security Requirements:
  - This system should be accessible only within the institute LAN
  - The database should not store any password in plain text - a hashed value has to be stored
- NFR3. Software Quality Attributes
- NFR4. Database Requirements

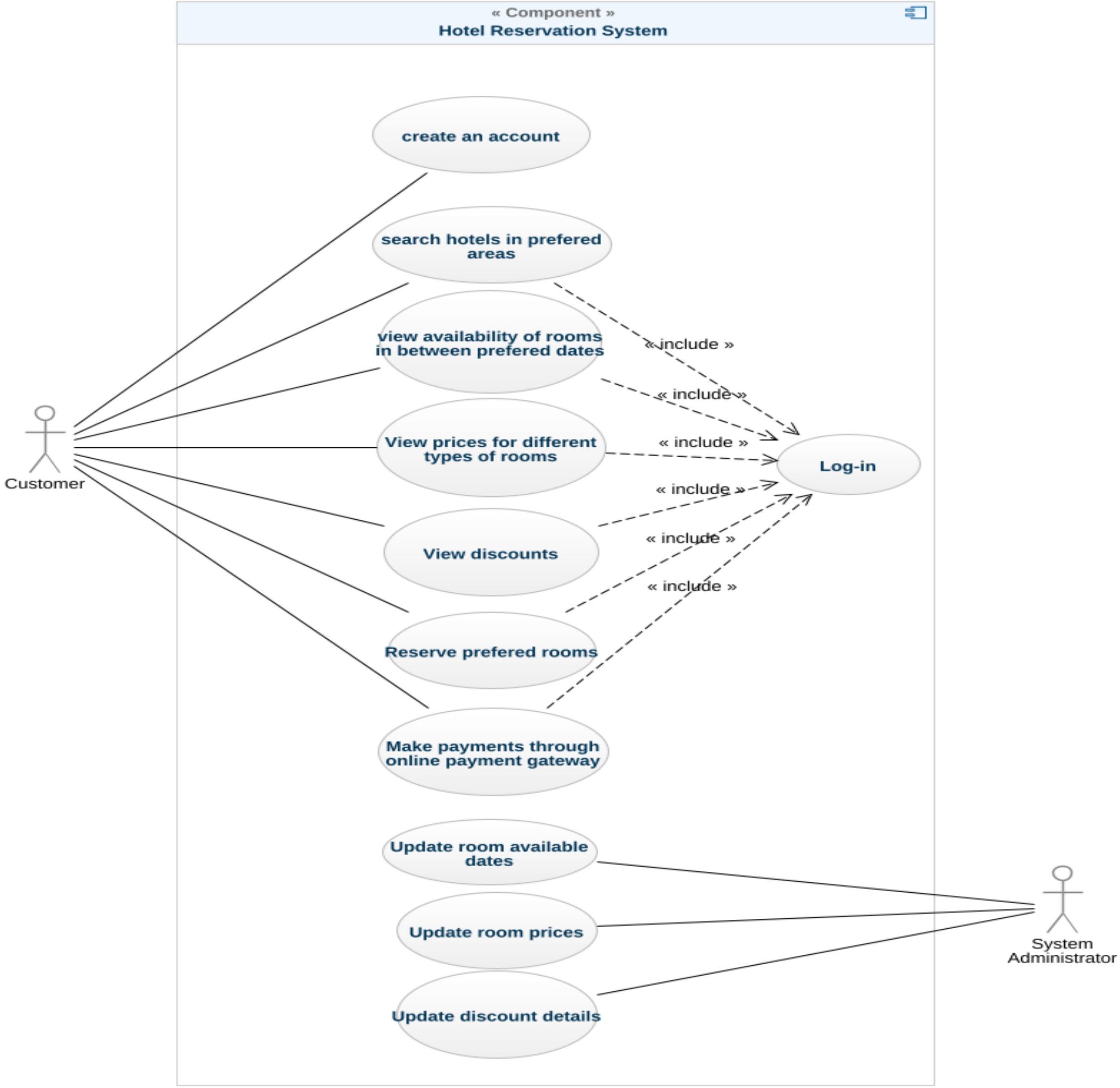
4. Risk

- **Schedule Risk**-Project schedules get slipped when project tasks and schedule release risks are not addressed properly.
- **Operational Risks**- Risk of loss due to improper process implementation, failed system or some external event risks.
- **Technical Risks**- Technical risks generally lead to failure of functionality and performance.

5. User stories:

- As a user I want to be able to be able to view all rooms in order to know which are available
- As a user I want to be able to search for all available rooms by location in order to easily see which rooms are available.
- As a user I want to be able to cancel room bookings in order to remove them if no longer required.

UML Diagram



Test suite:

Customer:

Test case-1: Login details.
Test case-2: Search availability.
Test case-3: Booking room.
Test case-4: checking reservation

Receptionist:

Test case-1: Update
Test case-2: Add.
Test case-3: Delete

Reservation

Test case-1: check for availability of rooms
Test case-2: display rate for all rooms
Test case-3: confirm or cancel the booking

4)E – TICKETING

1. Functional requirement

Client:

- The client enters destination and travel dates
- The client can choose the offers
- The client receives the e – ticket

Front desk manager:

- The front desk manager enters the passenger information
- The front desk manager confirm the payment and ticket
- The front desk manager updates the ticket status to client

Admin:

- The admin manages the flight schedules
- The admin manages the users.

Non Functional Requirements

- Performance Requirements:
  - This system should remain accessible 24x7
  - At least 50 users should be able to access the system altogether at any given time
- Security Requirements:
  - This system should be accessible only within the institute LAN
  - The database should not store any password in plain text - a hashed value has to be stored
- Software Quality Attributes
- Database Requirements

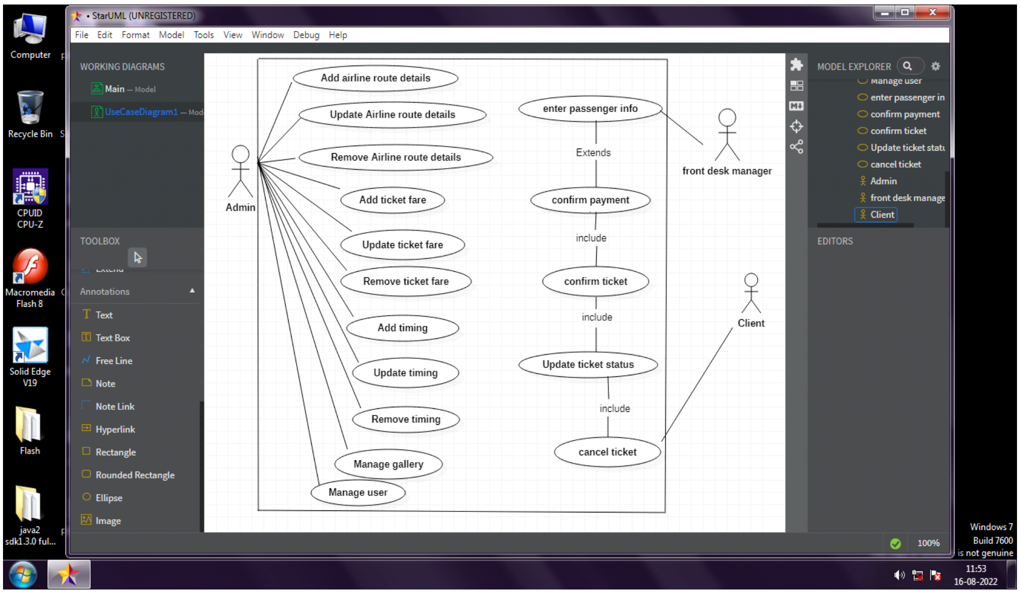
2. Risk

- **Schedule Risk**-Project schedules get slipped when project tasks and schedule release risks are not addressed properly.
- **Operational Risks**- Risk of loss due to improper process implementation, failed system or some external event risks.
- **Technical Risks**- Technical risks generally lead to failure of functionality and performance.

3. User stories

- As a user, I want the destination and travel dates so that I can book an e- ticket.
- As a front desk manager, I want the passenger information and payment so that I can confirm ticket.
- As a user, I want Cancellation option, so that I cancel the trip and get refund.

UML Diagram



1. <u>TEST SUITE</u>

Test suit 1: Admin

Test case1: Add airline details

Test case2: update airline route details

Test case3: remove airline route details

Test case4: add ticket fare

Test case5: update ticket fare

Test case6: remove ticket fare

Test suit 2: Front desk manager

Test case1: enter passenger info

Test suit 3 : client

Test case1: confirm payment

Test case2: confirm ticket

Test case3: update ticket status

Test case4: cancel ticket

5)Customer service information System

1. **Requirements :**

Functional requirements:

Centre agent:

- Centre agent will create new call details
- Save customer data
- Create work assignment letters
- Schedule the visit.

Technician:

- Technician will update status of work assignment letters
- order spare part
- check the fulfilment ordered spare part

Call centre :

- It will view the dashboard
- It will create the call reception report

Marketing manager:

- It will view the dashboard
- It will create the call reception report

Non – functional requirements :

Performance Requirements

NF1. Data in database should be updated within 2 seconds.
NF2. Query results must return results within 5 seconds
NF3. Load time of UI should not take more than 2 seconds
NF4. Login Validation should be done within 3 seconds
NF5. Response to customer inquiry must be done within 5 minutes.

Security Requirements

NF6. All external communications between the data's server and client must be encrypted
NF7. All data must be stored, protected or protectively marked.
NF8. Payment Process should use HTTP over Secure protocol to secure the payment transactions

Safety Requirements

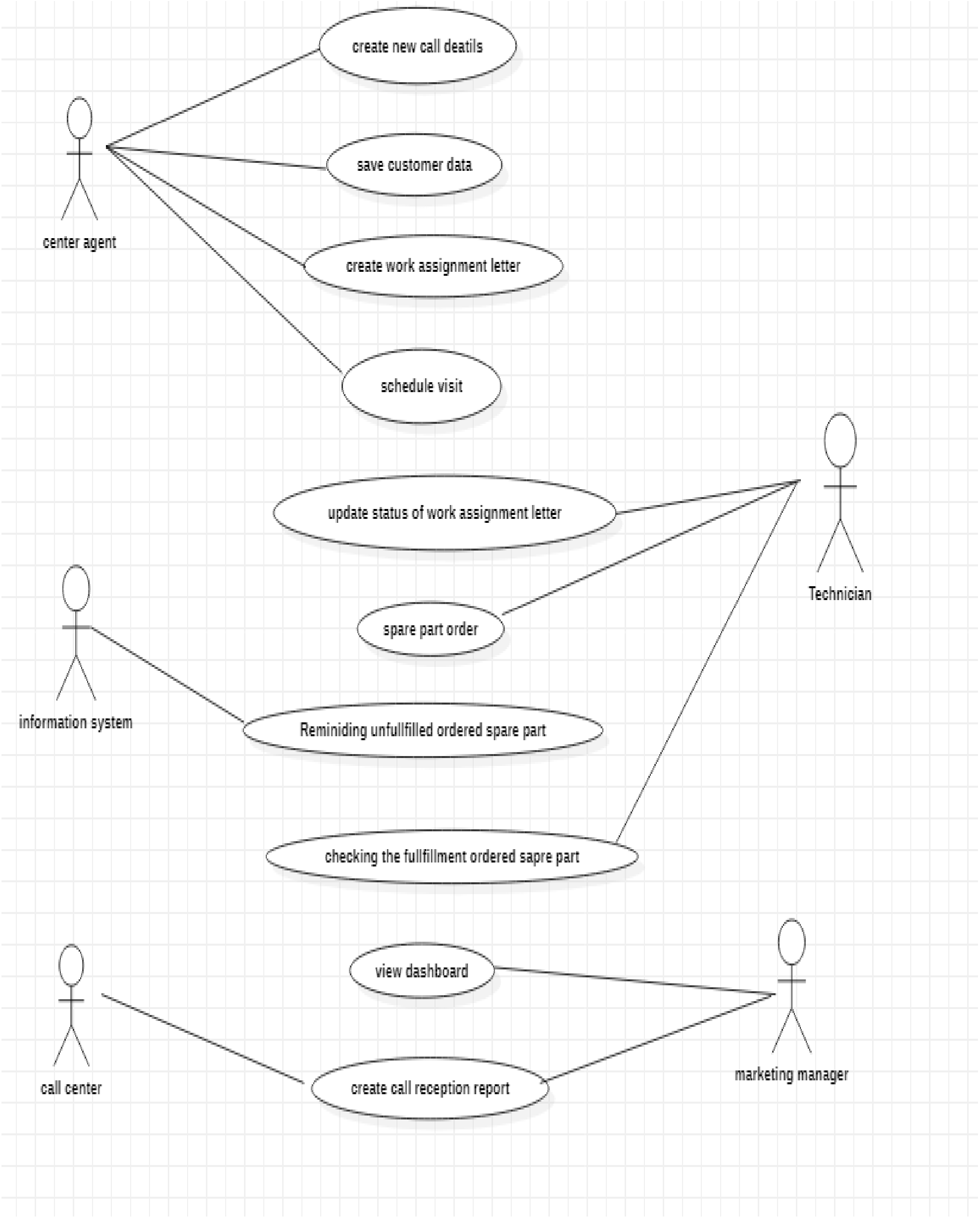
NF9. Database should be backed up every hour.
NF10. Under failure, system should be able to come back at normal operation under an hour

2. **Risks :**

- **Schedule Risk**-Project schedules get slipped when project tasks and schedule release risks are not addressed properly.
- **Operational Risks**- Risk of loss due to improper process implementation, failed system or some external event risks.
- **Technical Risks**- Technical risks generally lead to failure of functionality and performance.

3. **User stories :**
    - As a center Agent I want to create new call details to customer call service
    - As a technician I want to update status of work assignment letter and also spare part order to the customer service information system
    - As a marketing manager I want to view a dashboard and also create call reception report to the customer service information system

4. **UML Diagram :**


create new call deatils
center agent
save customer data
create work assignment letter
schedule visit
update status of work assignment letter
Technician
spare part order
information system
Reminiding unfullfilled ordered spare part
checking the fullfillment ordered sapre part
view dashboard
call center
marketing manager
create call reception report

5. **Test suite:**

Test suite 1 : Center Agent

Test case1: create new call details

Test case2: saves customer data

Test case3: create work assignment letters

Test case4: schedule visit

Test suite 2 : Technician

Test case1: update status of work assignment letters
Test case2: spare part order
Test case3: check the fulfilment ordered spare part

Test suite 3 : Information system

Test case1: Reminding unfulfilled ordered spare part

Test suite 4: call centre

Test case1: view dashboard
Test case2: create call reception report

Test suit 5 : marketing manager

Test case1: view dashboard
Test case2: create call reception report

6)Railway reservation system

1. **Requirements:**

Functional requirements:

- Admin and customer can Sign Up for the Account
- Admin and customer Sign In for the Reservation
- Name of the Customer
- Customer and Admin can see seats that's are not Booked
- Customer and Admin can see seats that's are already Booked
- Admin can Delete Reservation
- Customer can apply for refund
- Customer can see the Train timing
- Admin can See and change the train timing
- Verification of all these Information

Nonfunctional requirements:

Performance Requirements

NF1. Data in database should be updated within 2 seconds.
NF2. Query results must return results within 5 seconds
NF3. Load time of UI should not take more than 2 seconds
NF4. Login Validation should be done within 3 seconds
NF5. Response to customer inquiry must be done within 5 minutes.

Security Requirements

NF6. All external communications between the data's server and client must be encrypted
NF7. All data must be stored, protected or protectively marked.
NF8. Payment Process should use HTTP over Secure protocol to secure the payment transactions

Safety Requirements

NF9. Database should be backed up every hour.
NF10. Under failure, system should be able to come back at normal operation under an hour

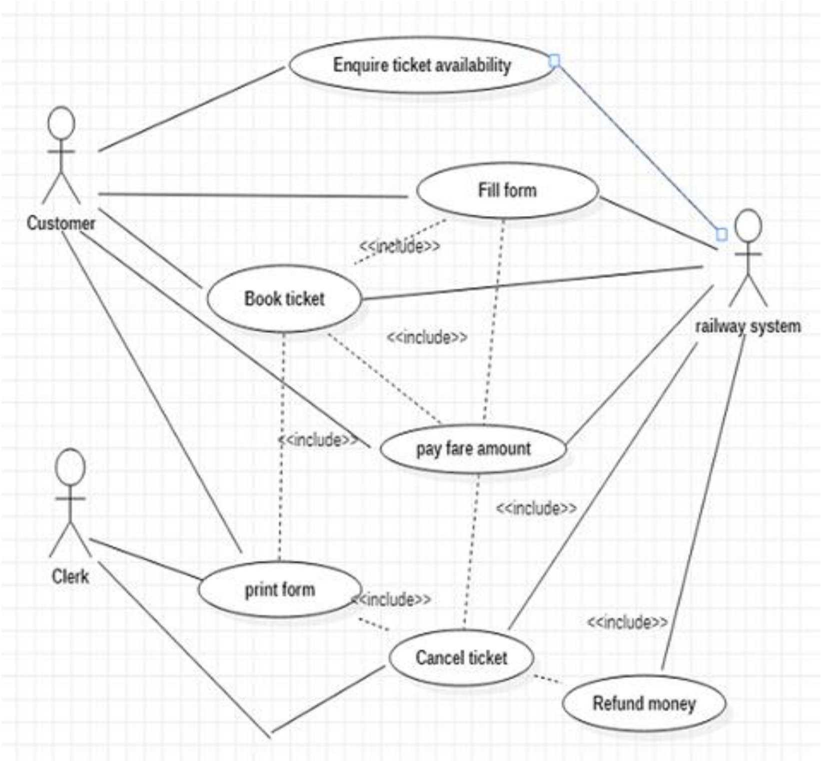
2. **Risks:**

- **Schedule Risk**-Project schedules get slipped when project tasks and schedule release risks are not addressed properly.
- **Operational Risks**- Risk of loss due to improper process implementation, failed system or some external event risks.
- **Technical Risks**- Technical risks generally lead to failure of functionality and performance.

3. **User story:**

- As a User I want to login So, that I can start further process
- As a User I want to complete the payment So, that I can have ticket confirmation.
  As a User I want to complete the transaction So, that I can start new transaction.

4. UML Diagram:



5. Test suite:

Test Suite: Customer

Test case 1: Enquire ticket availability
Test case 2: Fill form
Test case 3: Book ticket
Test case 4: pay fare amount
Test case 5: cancel ticket

Test Suite: Clerk

Test case 1: print form
Test case 2: cancel ticket

Test Suite: Railway station

Test case 1: Enquire ticket availability
Test case 2: Fill form
Test case 3: Book ticket
Test case 4: pay fare amount
Test case 5: print form
Test case 6: cancel ticket
Test case 7: Refund money

7) LIBRARY MANAGEMENT SYSTEM

1. FUNCTIONAL REQUIREMENTS

New user registration: Any member has to register himself with the Library Information System

Search book: A book could be searched by its:

- Title
- Authors name
- Publisher's name

User login: A registered user of can login to the system by providing his employee ID and password

Issue book: Any member of LIS can issue a book against his account provided that:

- The book is available in the library i.e. could be found by searching for it in LIS
- No other member has currently issued the book
- Current user has not issued the maximum number of books

Return book: A book once issued should be returned within the next 20 days

Reissue book: user might choose to reissue the book, and get the permission to keep it for another 20 days.

Non – functional requirements :

Performance Requirements

NF1. Data in database should be updated within 2 seconds.
NF2. Query results must return results within 5 seconds
NF3. Load time of UI should not take more than 2 seconds
NF4. Login Validation should be done within 3 seconds
NF5. Response to customer inquiry must be done within 5 minutes.

Security Requirements

NF6. All external communications between the data's server and client must be encrypted
NF7. All data must be stored, protected or protectively marked.
NF8. Payment Process should use HTTP over Secure protocol to secure the payment transactions

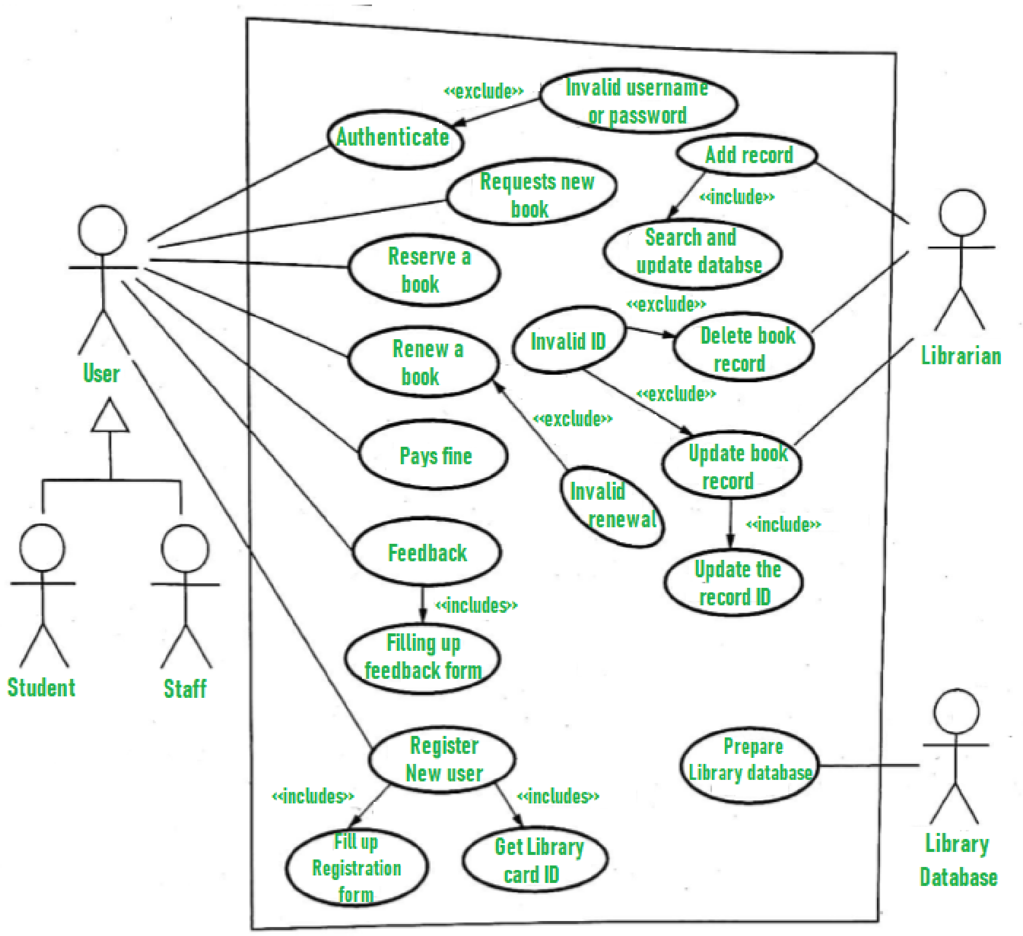
Safety Requirements

NF9. Database should be backed up every hour.
NF10. Under failure, system should be able to come back at normal operation under an hour

2. Risks :

- **Schedule Risk**-Project schedules get slipped when project tasks and schedule release risks are not addressed properly.
- **Operational Risks**- Risk of loss due to improper process implementation, failed system or some external event risks.
- **Technical Risks**- Technical risks generally lead to failure of functionality and performance.

UML DIAGRAM



4.

STUDENT
Test Case 1:-login
Test Case2:-Apply library card
Test Case3:-Renew a book
Test Case4:-Reserve a book
Test Case4:-pay fine
Test Case5:-Return book

LIBRARIAN
Test Case1:-Book information inquiry
Test Case2:-Update book information
Test Case3:-Process burrowing information

STAFF
Test Case1:-Data base update
Test case2:-System maintenance

8) Online Shopping

1.Functional requirements

Online Customer :-

- Online Customer will view the products
- Customer purchase the product
- This System should manage the customer information
- This System will create the product view

Marketing Administrator :-

- Marketing Administrator will manage products and report the product sales

Payment Processing System:-

- The Customer purchases the product and pay the payment

Warehouse Clerk:-

- The clerk will fulfill the orders

Warehouse Manager:-

- The manager can report the inventory and manage the inventory3.3

Non – functional requirements :

Performance Requirements

NF1. Data in database should be updated within 2 seconds.
NF2. Query results must return results within 5 seconds
NF3. Load time of UI should not take more than 2 seconds
NF4. Login Validation should be done within 3 seconds
NF5. Response to customer inquiry must be done within 5 minutes.

Security Requirements

NF6. All external communications between the data's server and client must be encrypted
NF7. All data must be stored, protected or protectively marked.
NF8. Payment Process should use HTTP over Secure protocol to secure the payment transactions

Safety Requirements

NF9. Database should be backed up every hour.
NF10. Under failure, system should be able to come back at normal operation under an hour

2. Risks :

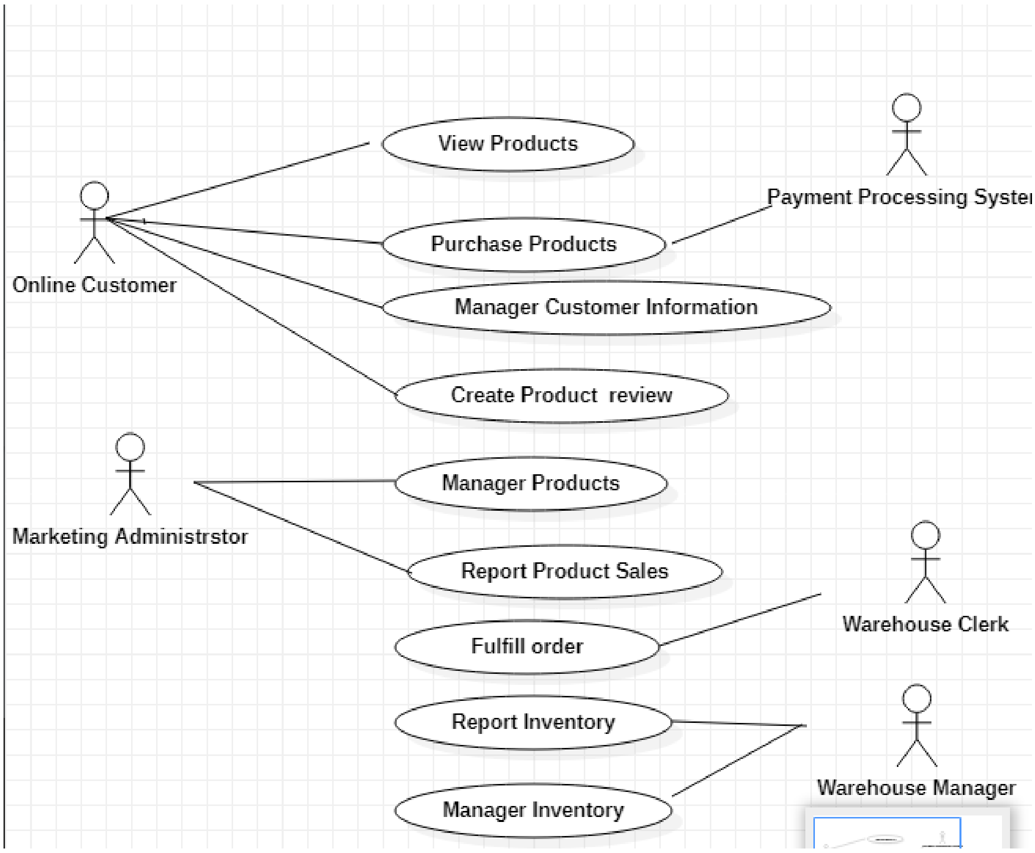
- **Schedule Risk**-Project schedules get slipped when project tasks and schedule release risks are not addressed properly.
- **Operational Risks**- Risk of loss due to improper process implementation, failed system or some external event risks.
- **Technical Risks**- Technical risks generally lead to failure of functionality and performance.

3.User Stories :-

- ✓ Customer :- As a Customer I need good customer care service to talk about ordered products
- ✓ Marketing Administrator :- As a Marketing Administrator I need a high populated Social media to promote our product

- ✓ Payment Processing System :- As a Admin I need a secure server to make the secure payments
- ✓ Warehouse Clerk :- As a Warehouse Clerk I need good Computer to Calculate the bills and orders
- ✓ Warehouse Manager :- As a Warehouse Manager I need a good AI [Artificial intelligence] robots to make fast packaging

UML (Use case Diagram):-



4.Test Suite :-

Online Customer
Test Case 1:- Login
Test Case 2:- View Products
Test Case 3:- Purchase Product
Test Case 4:- Manager Customer Information
Test Case 5:- Create Product Review

Marketing Administrator
Test Case 1:- Manage Product
Test Case 2:- Report Product Sales

Payment Processing System
Test Case 1:- Purchase Products

Warehouse Clerk
Test Case 1:- Fulfill Order

Warehouse Manager
Test Case 1:- Report Inventory
Test Case 2:- Manage Inventory

9)ELECTRONIC BILLING SYSTEM

1. Functional Requirements

- Login For Admin
- Forgot Password for Admin
- Edit profile For Admin

Manage Customer

- Adding New Customer
- Edit the Exiting Customer
- View Profile of the Customer
- Listing of all Customer

Manage Bills

- Add Payments of the Customer
- View Details of the Bills
- Listing of all Bills
- Filter Bills according to Customer

Manage Payments

- Report of all Customer
- Report of all Bills
- Report of all payments

Non – functional requirements :

Performance Requirements
NF1. Data in database should be updated within 2 seconds.
NF2. Query results must return results within 5 seconds
NF3. Load time of UI should not take more than 2 seconds
NF4. Login Validation should be done within 3 seconds
NF5. Response to customer inquiry must be done within 5 minutes.

Security Requirements
NF6. All external communications between the data's server and client must be encrypted
NF7. All data must be stored, protected or protectively marked.
NF8. Payment Process should use HTTP over Secure protocol to secure the payment transactions

Safety Requirements
NF9. Database should be backed up every hour.
NF10. Under failure, system should be able to come back at normal operation under an hour

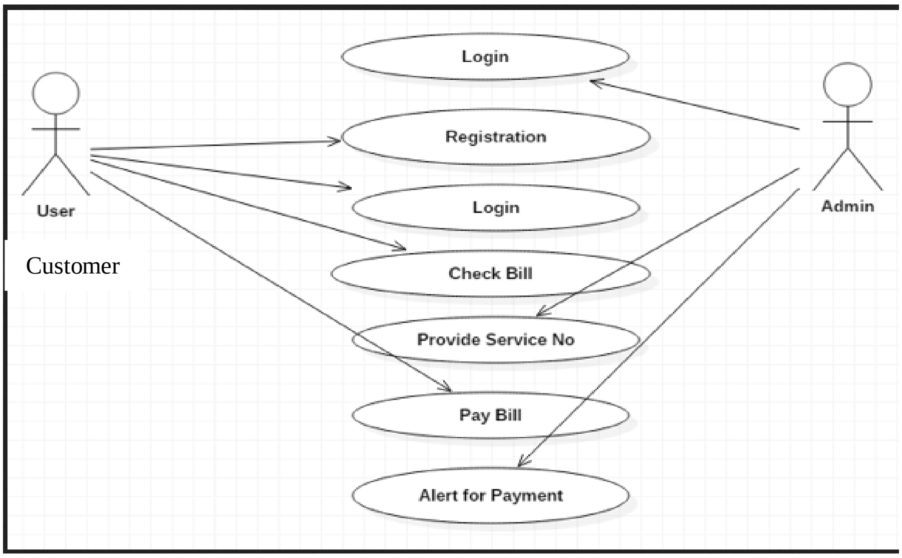
2.Risks :

- **Schedule Risk**-Project schedules get slipped when project tasks and schedule release risks are not addressed properly.
- **Operational Risks**- Risk of loss due to improper process implementation, failed system or some external event risks.
- **Technical Risks**- Technical risks generally lead to failure of functionality and performance.

3.User Stories :-

- As a user I want to be able to pay for my bill online using my credit/debit card in order to complete my payment.
- As a user I want to be able to pay for my bill using cash (in person) in order to complete my payment.
- As a Admin I need a secure server to make the secure payments.

UML (Use case Diagram):-



4.Test Suite :-

Admin

Test Case 1:- Login
Test Case 2:- Forgot Password
Test Case 3:- Edit profile

Manage Customer

Test Case 1:- Adding New Customer
Test Case 2:- Edit the Exiting Customer
Test Case 3:- View Profile of the Customer
Test Case 4:- Listing of all Customer

Manage Bills

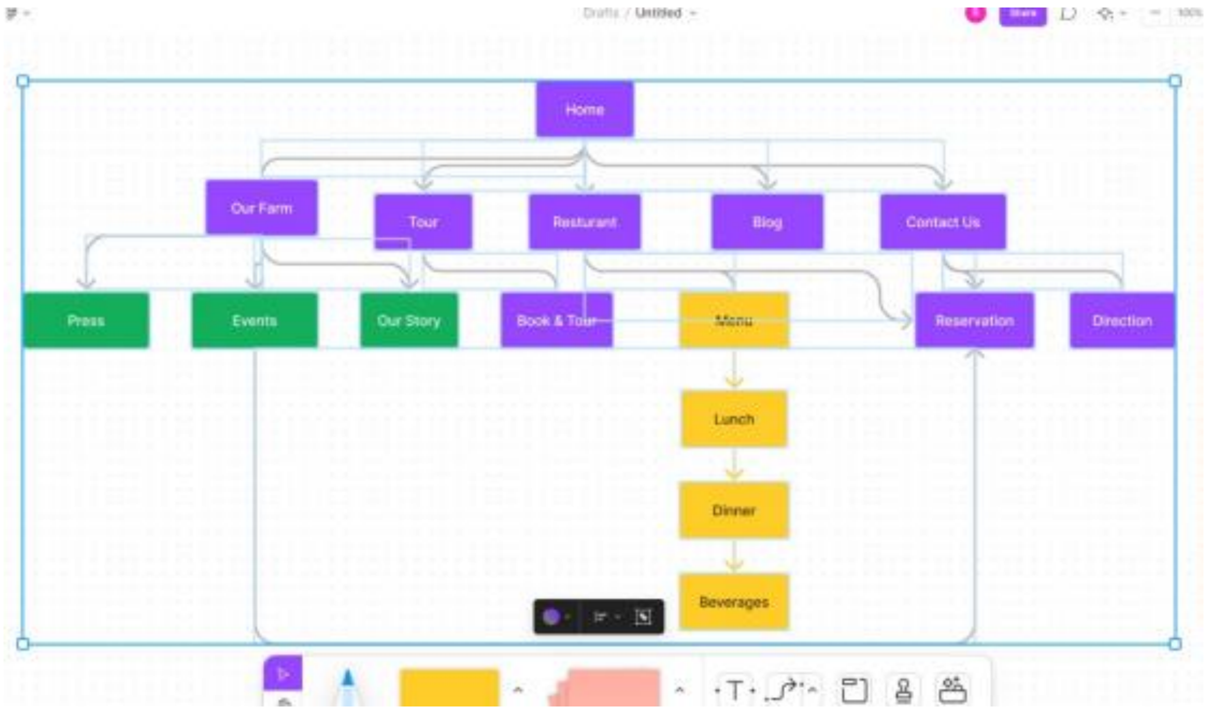
Test Case 1:- Add Payments of the Customer
Test Case 2:- View Details of the Bills
Test Case 3:- Listing of all Bills
Test Case 4:- Filter Bills according to Customer

Manage Payments

Test Case 1:- Report of all Customer
Test Case 2:- Report of all Bills
Test Case 3:- Report of all payments

10.Create sitemap and wireframe for above created user stories Create Sitemap wireframe for above created user stories.(ANY project)

(Tools such as sketch, Adobe XD, Figma, etc. can be used. NOTE: Download any of the tool mentioned in the Title.

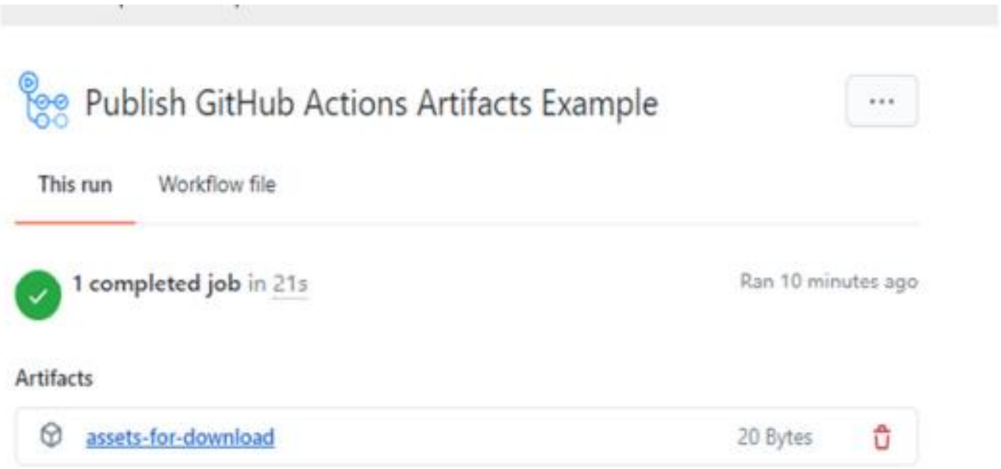


Step-1: Download the Figma Application Through Any browser,
Step-2: Now install the application in the desktop
Step-3: Open the application and create an account using Google account
Step-4: Now again Login into Figma application and create a new page using new figjam
, Step-5: Using Clipboard, click on ellipse and choose it as square,
 Step-6: Make some clips by pulling blocks from ellipse, Step-7: arrange the blocks as shown in the above diagram.

11. Create Git (similar tool) account and configure repository Steps to publish Git artifacts A developer should follow these five steps to publish GitHub Actions artifacts for download:

1. Perform Git Actions build steps
2. Create a temporary folder in the container being used
3. Copy all artifacts of interest into that temporary folder
4. Use GitHub's upload-artifact action 1. Provide a meaningful name for the artifact download link 2. Specify the path to the folder containing your GitHub Action artifacts

5. Run the GitHub Actions workflow and find the published artifacts on the workflow's build page
6. The easiest way to demonstrate how GitHub's artifact upload action works is to add a step to a simple workflow that creates a temporary directory. Then, use the touch and echo commands to create a few simple files. Once a developer completes this action, the files will publish as artifacts. Published artifacts in GitHub If a developer isn't familiar with the echo and output switch, the following command will create a file on the local filesystem named alpha.html with the text 'alpha' contained within it:



Publish GitHub Actions Artifacts Example

1 completed job in 21s

Ran 10 minutes ago

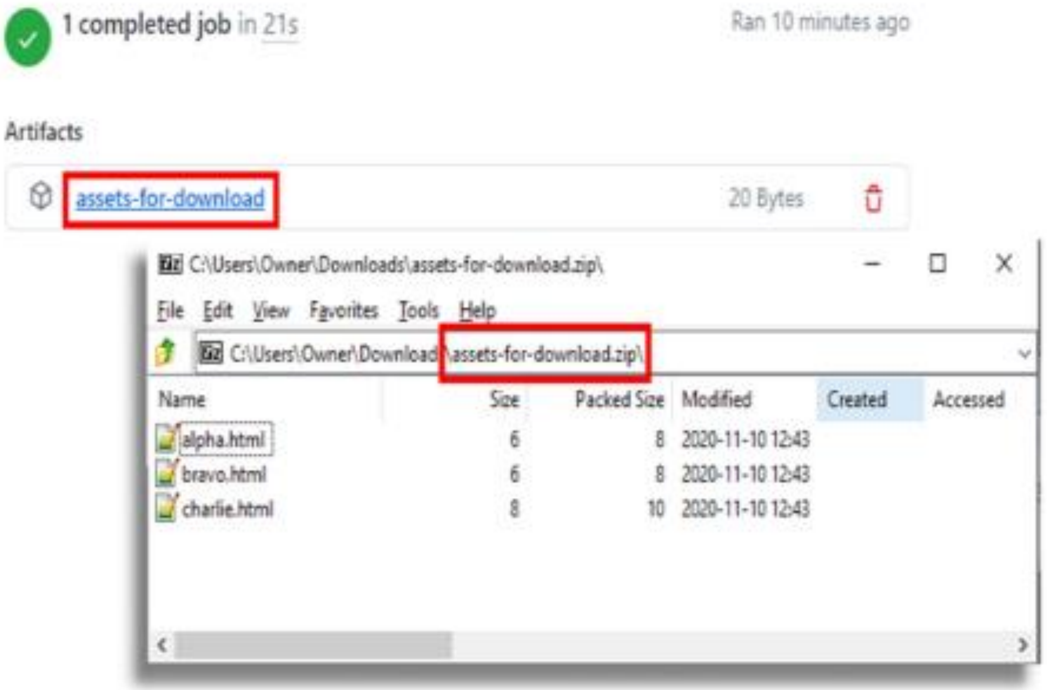
Artifacts

assets-for-download 20 Bytes

When you publish a GitHub Action artifact, it can be downloaded from the build's workflow summary page.

Workflow artifact downloads

- When this build runs, the status page of the workflow will include a link to download a file named assets-for-download.
- zip, which will contain the three files named alpha.html, bravo.html and charlie.html.
- This proves that the script works and makes the GitHub Action artifacts available for download.



1 completed job in 21s

Ran 10 minutes ago

Artifacts

assets-for-download 20 Bytes

C:\Users\Owner\Downloads\assets-for-download.zip\

C:\Users\Owner\Downloads\assets-for-download.zip\

Name	Size	Packed Size	Modified	Created	Accessed
alpha.html	6	8	2020-11-10 12:43		
bravo.html	6	8	2020-11-10 12:43		
charlie.html	8	10	2020-11-10 12:43		

12.Learn version control and configuration management with GIT: Create a repository project

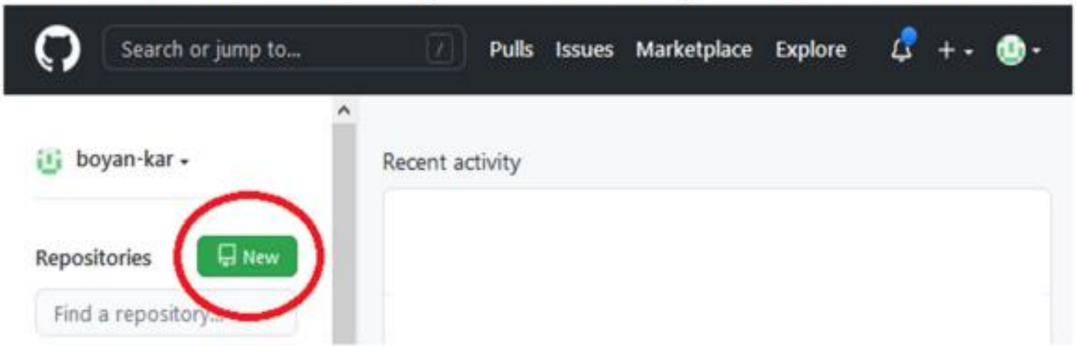
Step 1:On your computer, you need to install Git first. The process will depend on your operating system: please follow the instructions below by clicking the relevant button.

What is a repository?

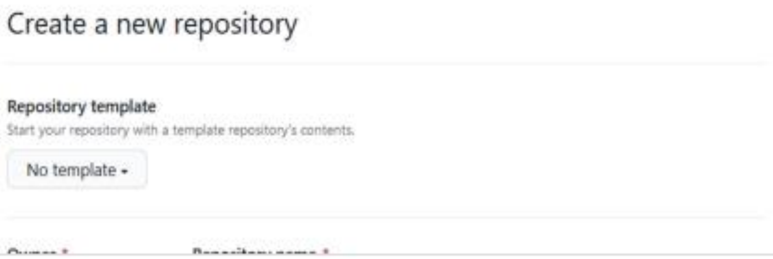
You can think of a repository (aka a repo) as a “main folder”, everything associated with a specific project should be kept in a repo for that project. Repos can have folders within them, or just be separate files.

2.Create your own repository and project folder structure

name that has no spaces or funky characters in it.

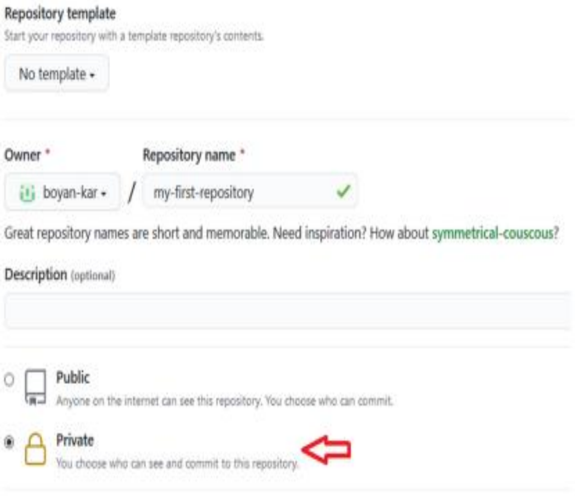


Step 3:Let’s create a new private repository. You can call it whatever you like if the name is available.


Create a new repository

Repository template
Start your repository with a template repository's contents.
No template

Step 3:Let’s create a new private repository. You can call it whatever you like if the name is available.



Repository template

Start your repository with a template repository's contents.

No template ▾

Owner * / Repository name *

boyan-kar ▾ / my-first-repository ✓

Great repository names are short and memorable. Need inspiration? How about symmetrical-couscous?

Description (optional)

○ Public

Anyone on the internet can see this repository. You choose who can commit.

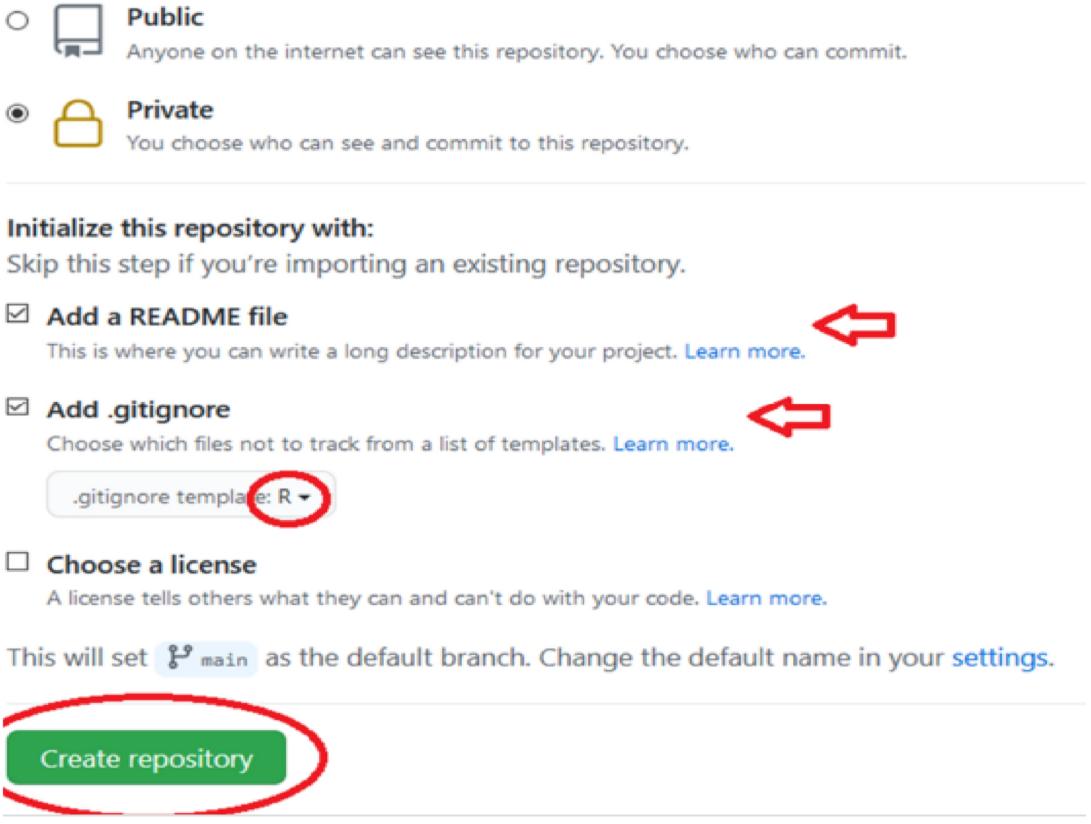
● Private

You choose who can see and commit to this repository.

Step 4:Click on Initialise repo with a README.md file. It’s common practice for each repository to have a README.md file,

Step 5:We will also create a .gitignore file. This file lets Git know what kind of files should not be included in the repository.

Once you are ready, click on Create repository



○ Public

Anyone on the internet can see this repository. You choose who can commit.

● Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you’re importing an existing repository.

☑ Add a README file

This is where you can write a long description for your project. Learn more.

☑ Add .gitignore

Choose which files not to track from a list of templates. Learn more.

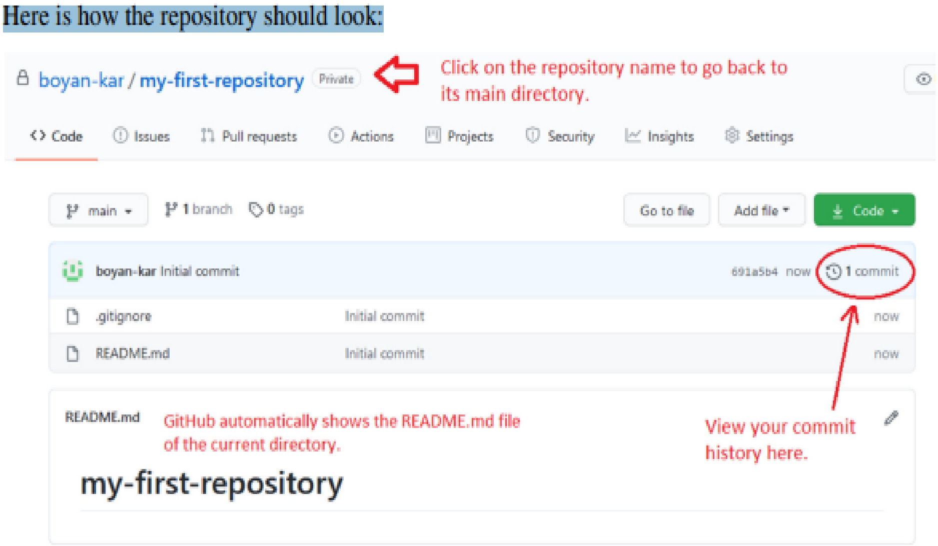
.gitignore template: R ▾

☐ Choose a license

A license tells others what they can and can't do with your code. Learn more.

This will set `main` as the default branch. Change the default name in your settings.

Here is how the repository should look:



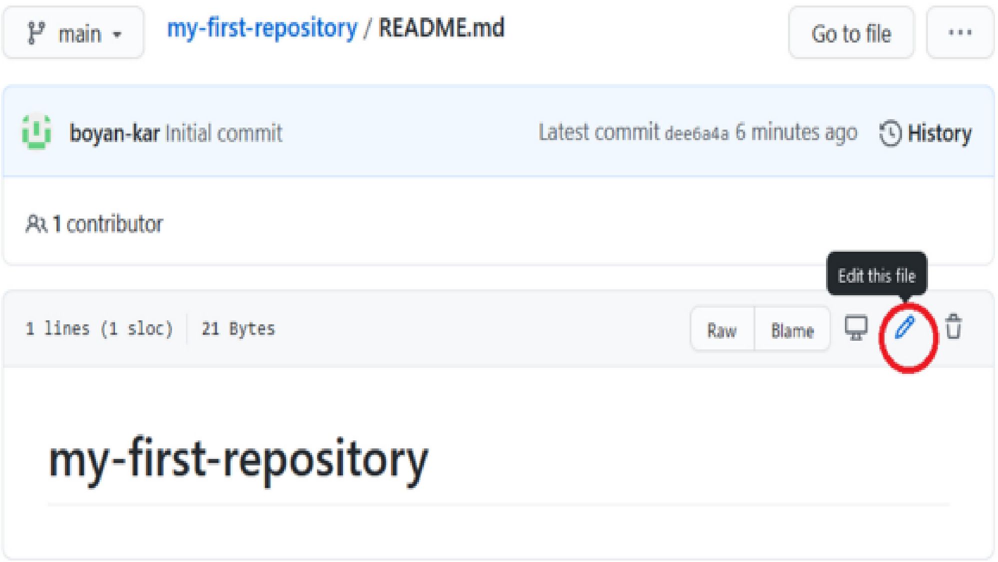
Here is how the repository should look:

Click on the repository name to go back to its main directory.

GitHub automatically shows the README.md file of the current directory.

View your commit history here.

You can directly edit your README.md file on Github by clicking on the file and then selecting Edit this file.

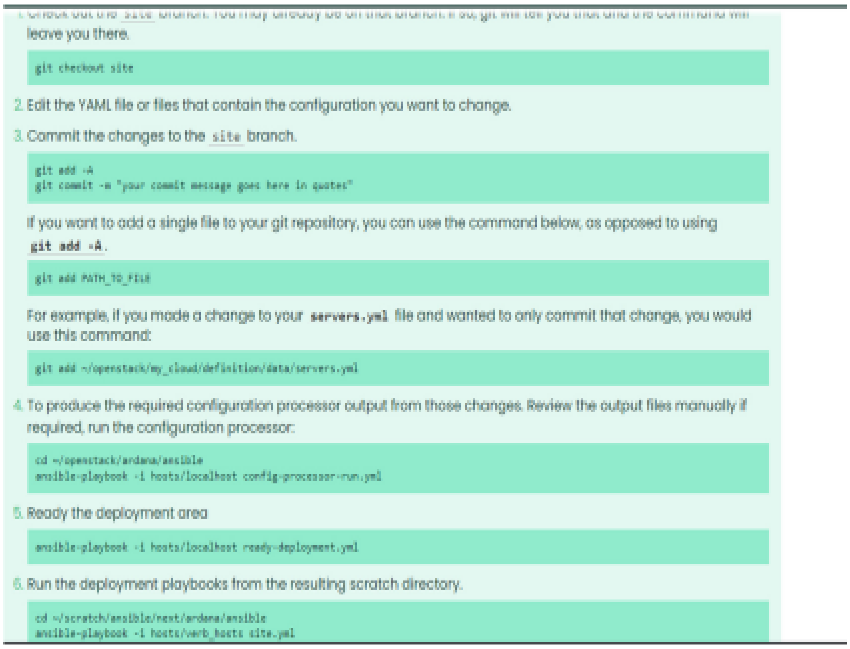


13.Configuration management with GIT: Using GIT for Configuration Management

Step 1:Initialization on a new deployment

Step 2: Updating any configuration, including the default configuration:

Step 3: Resolving Git merge conflicts



1. Check out the site branch. You may already be on that branch. If so, git will tell you that and the command will leave you there.

```
git checkout site
```

2. Edit the YAML file or files that contain the configuration you want to change.
3. Commit the changes to the site branch.

```
git add -A
git commit -m "your commit message goes here in quotes"
```

If you want to add a single file to your git repository, you can use the command below, as opposed to using `git add -A`.

```
git add PATH_TO_FILE
```

For example, if you made a change to your `servers.yml` file and wanted to only commit that change, you would use this command:

```
git add ~/openstack/my_cloud/definition/data/servers.yml
```

4. To produce the required configuration processor output from those changes. Review the output files manually if required, run the configuration processor:

```
cd ~/openstack/ardana/ansible
ansible-playbook -i hosts/localhost config-processor-run.yml
```

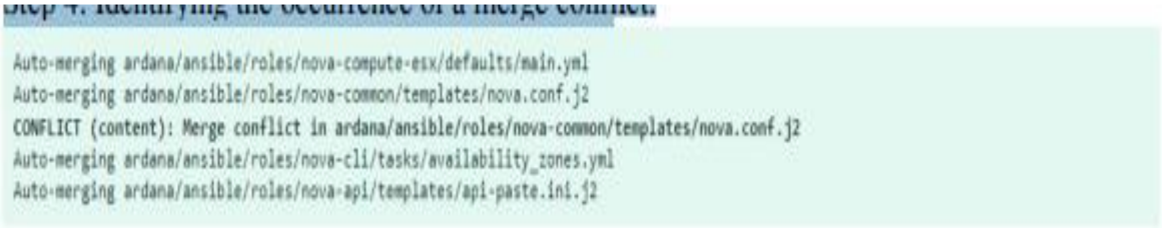
5. Ready the deployment area

```
ansible-playbook -i hosts/localhost ready-deployment.yml
```

6. Run the deployment playbooks from the resulting scratch directory.

```
cd ~/scratch/ansible/next/ardana/ansible
ansible-playbook -i hosts/verb_hosts site.yml
```

Step 4: Identifying the occurrence of a merge conflict:



Step 4: Identifying the occurrence of a merge conflict:

```
Auto-merging ardana/ansible/roles/nova-compute-esx/defaults/main.yml
Auto-merging ardana/ansible/roles/nova-common/templates/nova.conf.j2
CONFLICT (content): Merge conflict in ardana/ansible/roles/nova-common/templates/nova.conf.j2
Auto-merging ardana/ansible/roles/nova-cli/tasks/availability_zones.yml
Auto-merging ardana/ansible/roles/nova-api/templates/api-paste.ini.j2
```

Step 5: Examining Conflicts

Step 6: Examining differences between your current version and the previous upstream version

• The previous "upstream" version on the ardana branch. Your current version on the site branch. He new "upstream" version on the ardana branch

Step 7: Using stage markers to view clean versions of files (without conflict markers)

Step 8: Resolving the conflict There are two approaches to resolving the conflict:

1. Edit the merged file containing the conflict markers, keeping the change you want to preserve and removing the conflict markers and any changes you want to discard.
2. Take the new upstream version of the file and re-apply any changes you would like to keep from your current version.

Step 9: Resolving the conflict - editing the file containing the conflict markers

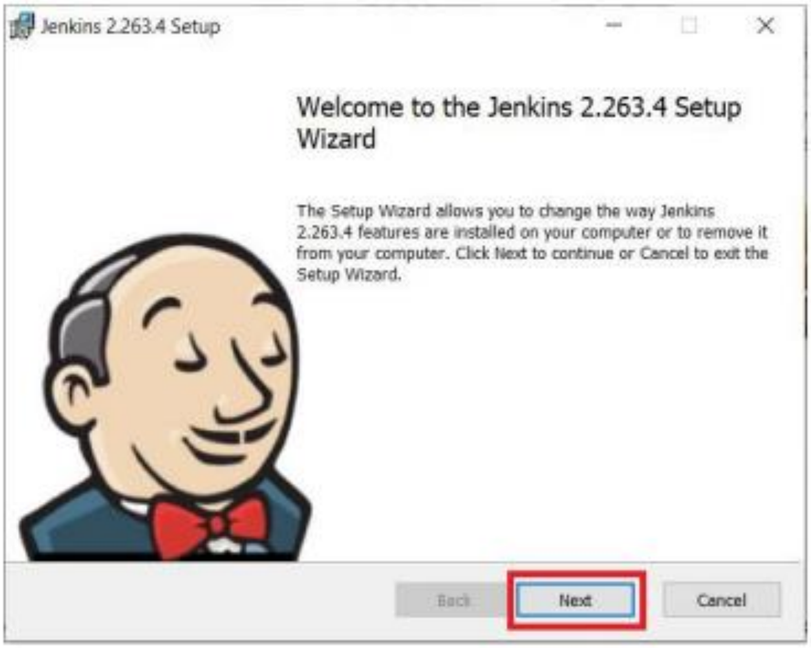
Step 10: Resolving the conflict - re-applying your changes to new upstream version

Step 11: Completing the merge procedure

Step 12: Recovering from Errors

14.Install and configure Jenkins Step

1: Setup wizard On opening the Windows Installer, an Installation Setup Wizard appears, Click Next on the Setup Wizard to start your installation.

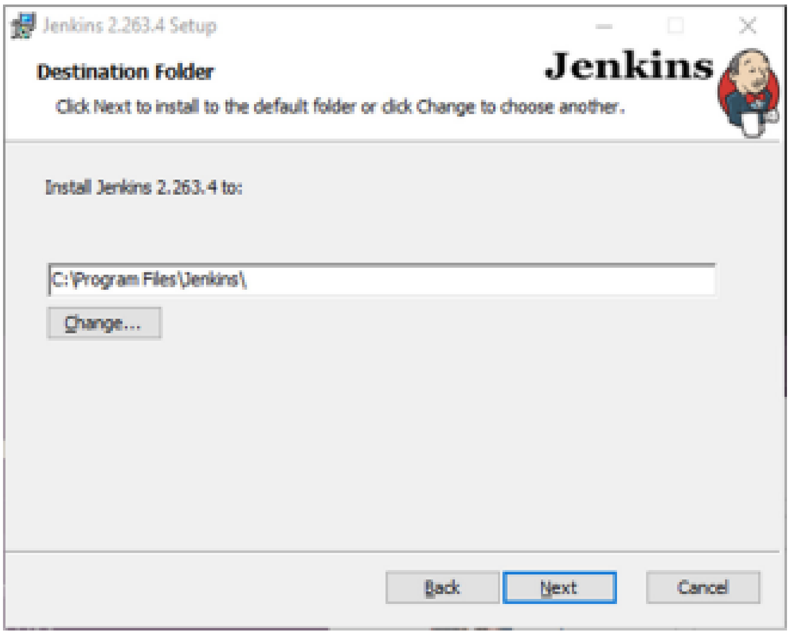


Jenkins 2.263.4 Setup

Welcome to the Jenkins 2.263.4 Setup Wizard

The Setup Wizard allows you to change the way Jenkins 2.263.4 features are installed on your computer or to remove it from your computer. Click Next to continue or Cancel to exit the Setup Wizard.

Step 2: **Select destination folder**
Select the destination folder to store your Jenkins Installation and click Next to continue.



Jenkins 2.263.4 Setup

Destination Folder

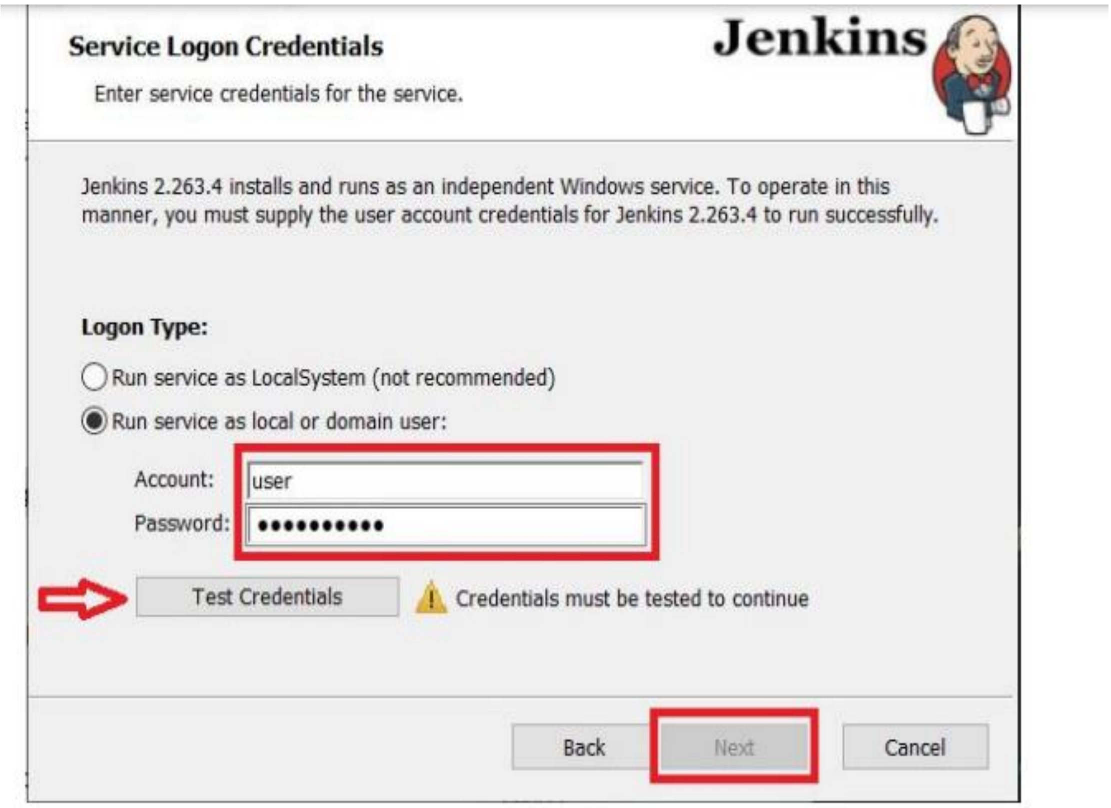
Click Next to install to the default folder or click Change to choose another.

Install Jenkins 2.263.4 to:

C:\Program Files\Jenkins\

Step 3: Service logon credentials

When Installing Jenkins, it is recommended to install and run Jenkins as an independent windows service using a local or domain user as it is much safer than running Jenkins using LocalSystem(Windows equivalent of root) which will grant Jenkins full access to your machine and services. To run Jenkins service using a local or domain user, specify the domain user name and password with which you want to run Jenkins, click on Test Credentials to test your domain credentials and click on Next.



Service Logon Credentials

Enter service credentials for the service.

Jenkins 2.263.4 installs and runs as an independent Windows service. To operate in this manner, you must supply the user account credentials for Jenkins 2.263.4 to run successfully.

Logon Type:

○ Run service as LocalSystem (not recommended)

● Run service as local or domain user:

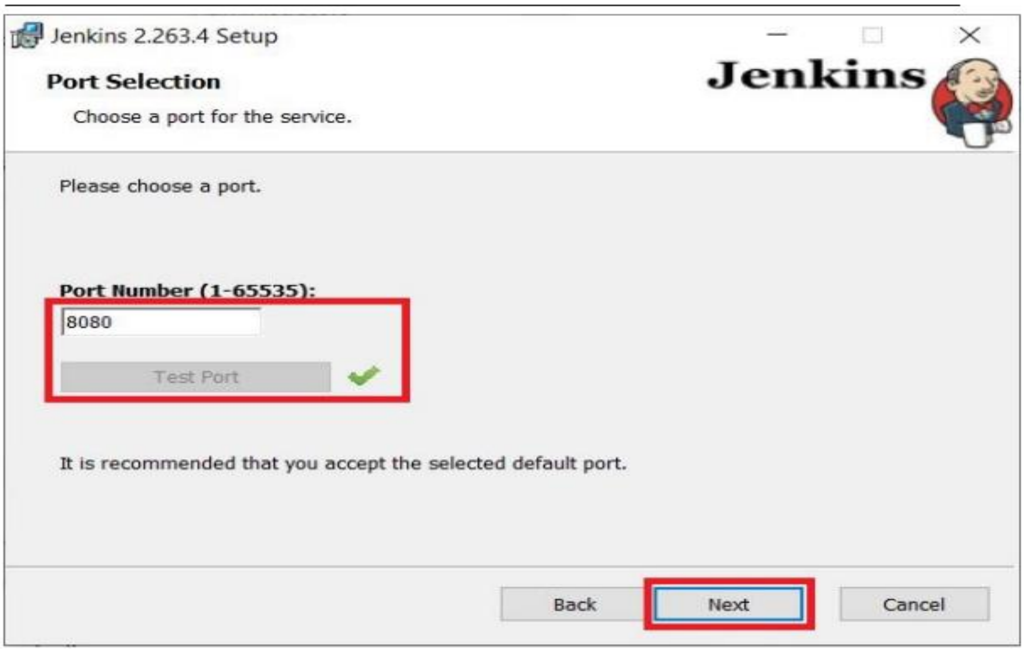
Account: user

Password: ••••••••••

Credentials must be tested to continue

Step 4: **Port selection**

Specify the port on which Jenkins will be running, Test Port button to validate whether the specified port if free on your machine or not. Consequently, if the port is free, it will show a green tick mark as shown below, then click on Next.



Jenkins 2.263.4 Setup

Port Selection

Choose a port for the service.

Please choose a port.

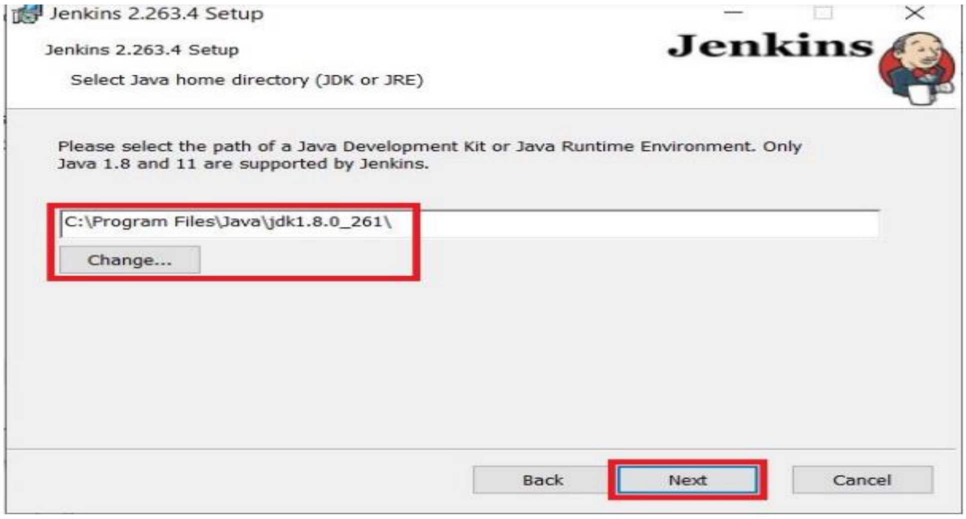
Port Number (1-65535): 8080

It is recommended that you accept the selected default port.

Step 5: Select Java home directory

The installation process checks for Java on your machine and prefills the dialog with the Java home directory. If the needed Java version is not installed on your machine, you will be prompted to install it.

Once your Java home directory has been selected, click on Next to continue.



Jenkins 2.263.4 Setup

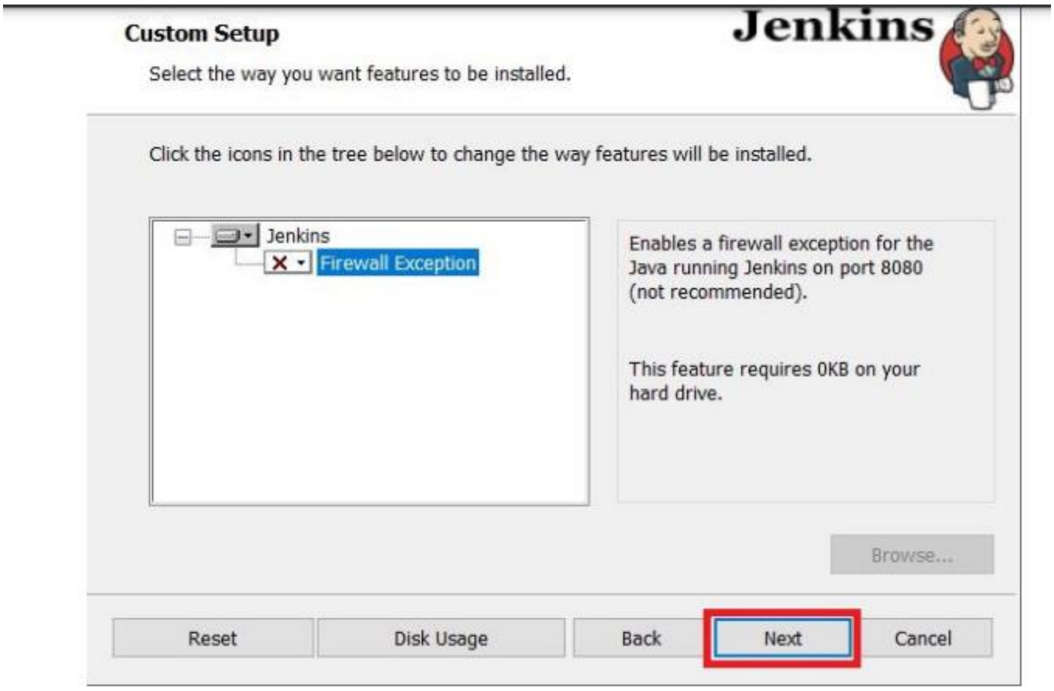
Jenkins 2.263.4 Setup

Select Java home directory (JDK or JRE)

Please select the path of a Java Development Kit or Java Runtime Environment. Only Java 1.8 and 11 are supported by Jenkins.

C:\Program Files\Java\jdk1.8.0_261\

Step 6: Custom setup Select other services that need to be installed with Jenkins and click on Next



Custom Setup

Select the way you want features to be installed.

Click the icons in the tree below to change the way features will be installed.

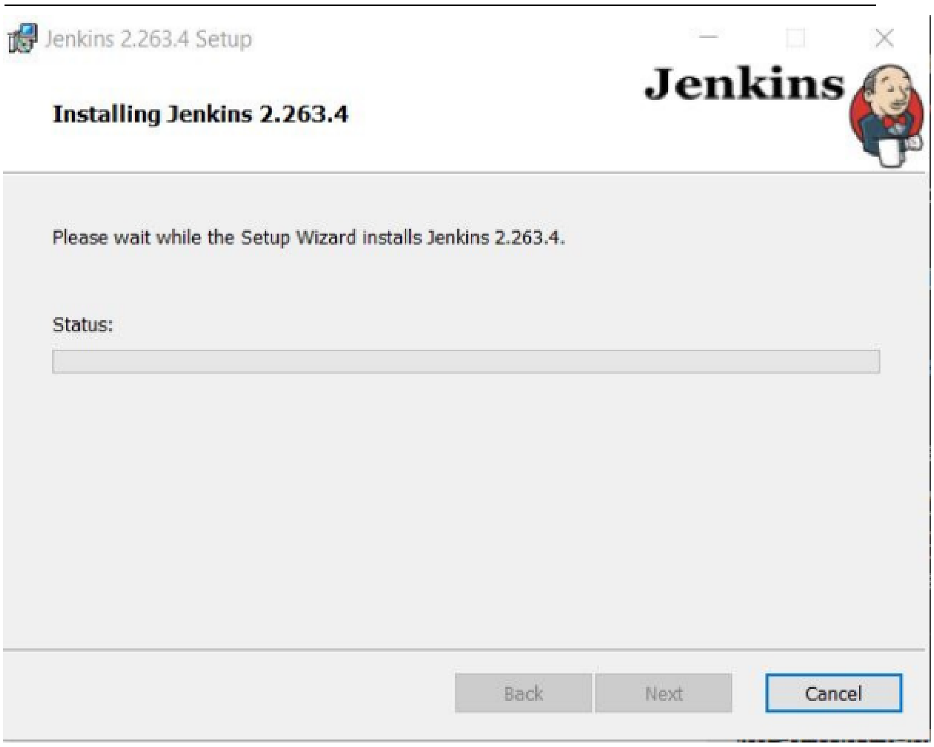
Enables a firewall exception for the Java running Jenkins on port 8080 (not recommended).

This feature requires 0KB on your hard drive.

Step 7**: Install Jenkins**

Click on the Install button to start the installation of Jenkins.

Additionally, clicking on the Install button will show the progress bar of installation, as shown below:



Jenkins 2.263.4 Setup

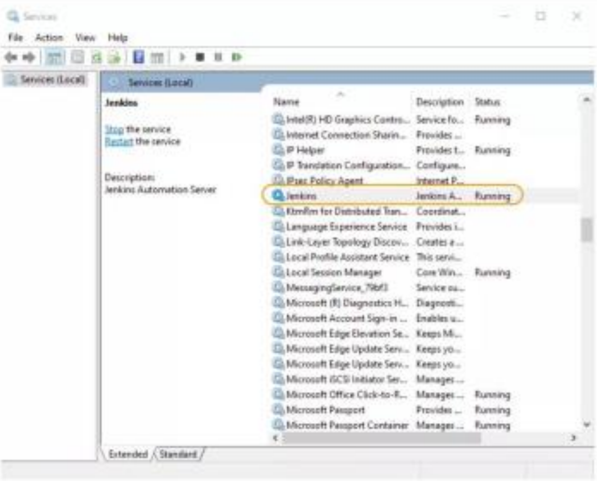
Installing Jenkins 2.263.4

Please wait while the Setup Wizard installs Jenkins 2.263.4.

Status:

Step 8: Finish Jenkins installation

Once the installation completes, click on Finish to complete the installation. Jenkins will be installed as a Windows Service. You can validate this by browsing the services section, as shown below:



Jenkins

Stop the service
Restart the service

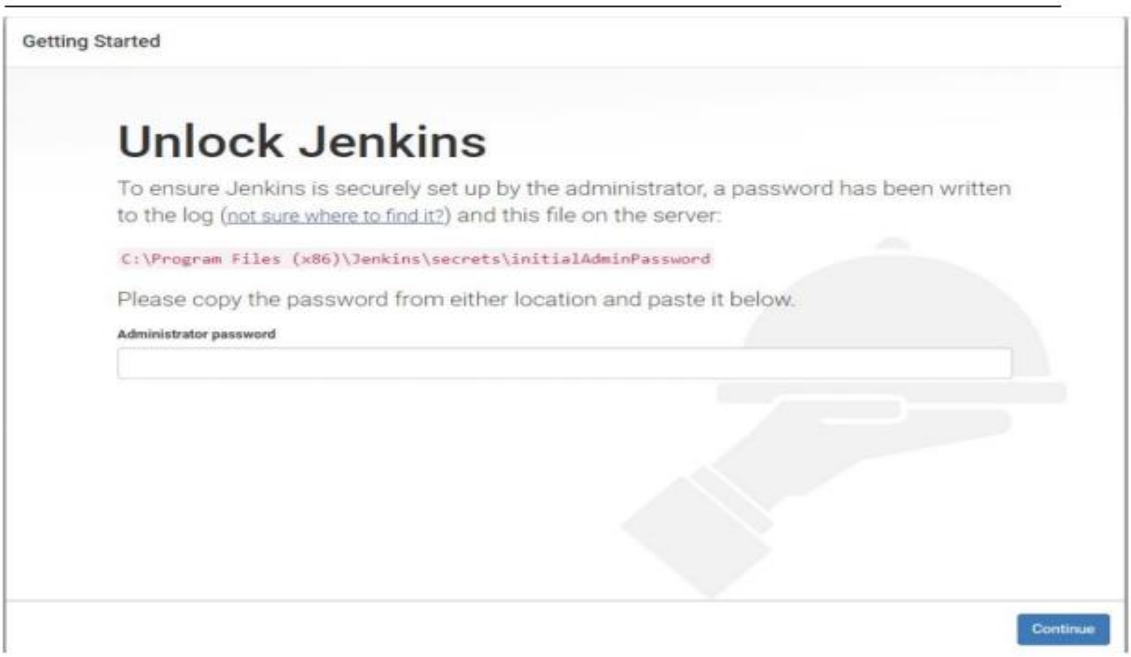
Description:
Jenkins Automation Server

Name	Description	Status
Intel(R) HD Graphics Contro...	Service fo...	Running
Internet Connection Sharin...	Provides ...	
IP Helper	Provides t...	Running
IP Translation Configuration...	Configure...	
IPsec Policy Agent	Internet P...	
Jenkins	Jenkins A...	Running
KtmRm for Distributed Tran...	Coordinat...	
Language Experience Service	Provides i...	
Link-Layer Topology Discov...	Creates a ...	
Local Profile Assistant Service	This servi...	
Local Session Manager	Core Win...	Running
MessagingService_79bf3	Service su...	
Microsoft (R) Diagnostics H...	Diagnosti...	
Microsoft Account Sign-in ...	Enables u...	
Microsoft Edge Elevation Se...	Keeps Mi...	
Microsoft Edge Update Serv...	Keeps yo...	
Microsoft Edge Update Serv...	Keeps yo...	
Microsoft iSCSI Initiator Ser...	Manages ...	
Microsoft Office Click-to-R...	Manages ...	Running
Microsoft Passport	Provides ...	Running
Microsoft Passport Container	Manages ...	Running

Post-installation setup wizard After downloading
, installing and running Jenkins, the post-installation setup wizard begins. This setup wizard takes you through a few quick "one-off" steps to unlock Jenkins, customize it with plugins and create the first administrator user through which you can continue accessing Jenkins.

15.Unlocking Jenkins

When you first access a new Jenkins instance, you are asked to unlock it using an automatically-generated password.
Step 1 Browse to http://localhost:8080 (or whichever port you configured for Jenkins when installing it) and wait until the Unlock Jenkins page appears.



Getting Started

Unlock Jenkins

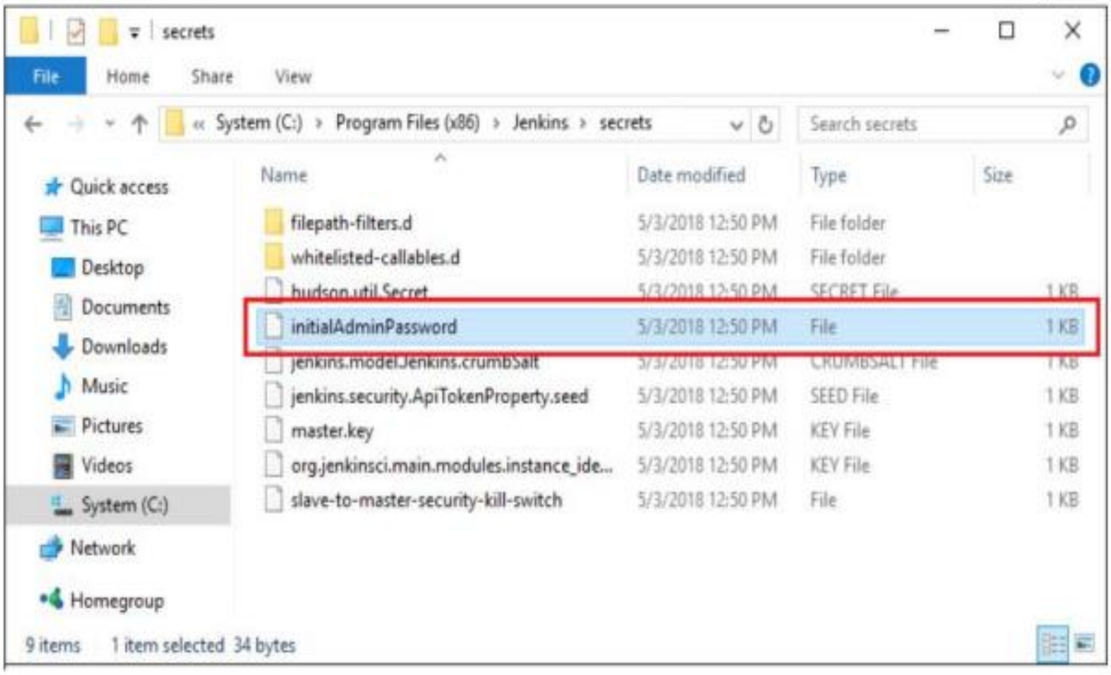
To ensure Jenkins is securely set up by the administrator, a password has been written to the log (not sure where to find it?) and this file on the server:

`C:\Program Files (x86)\Jenkins\secrets\initialAdminPassword`

Please copy the password from either location and paste it below.

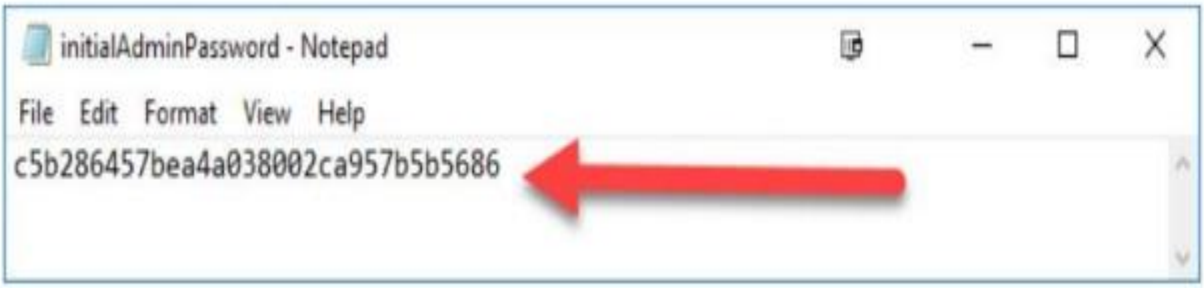
Administrator password

Step 2 The initial Administrator password should be found under the Jenkins installation path (set at Step 2 in Jenkins Installation).
For default installation location to C:\Program Files\Jenkins, a file called initialAdminPassword can be found under C:\Program Files\Jenkins\secrets.
However, If a custom path for Jenkins installation was selected, then you should check that location for initialAdminPassword file.



Name	Date modified	Type	Size
filepath-filters.d	5/3/2018 12:50 PM	File folder	
whitelisted-callables.d	5/3/2018 12:50 PM	File folder	
hudson.util.Secret	5/3/2018 12:50 PM	SECRET File	1 KB
initialAdminPassword	5/3/2018 12:50 PM	File	1 KB
jenkins.model.Jenkins.crumbSalt	5/3/2018 12:50 PM	CRUMBSALT File	1 KB
jenkins.security.ApiTokenProperty.seed	5/3/2018 12:50 PM	SEED File	1 KB
master.key	5/3/2018 12:50 PM	KEY File	1 KB
org.jenkinsci.main.modules.instance_ide...	5/3/2018 12:50 PM	KEY File	1 KB
slave-to-master-security-kill-switch	5/3/2018 12:50 PM	File	1 KB

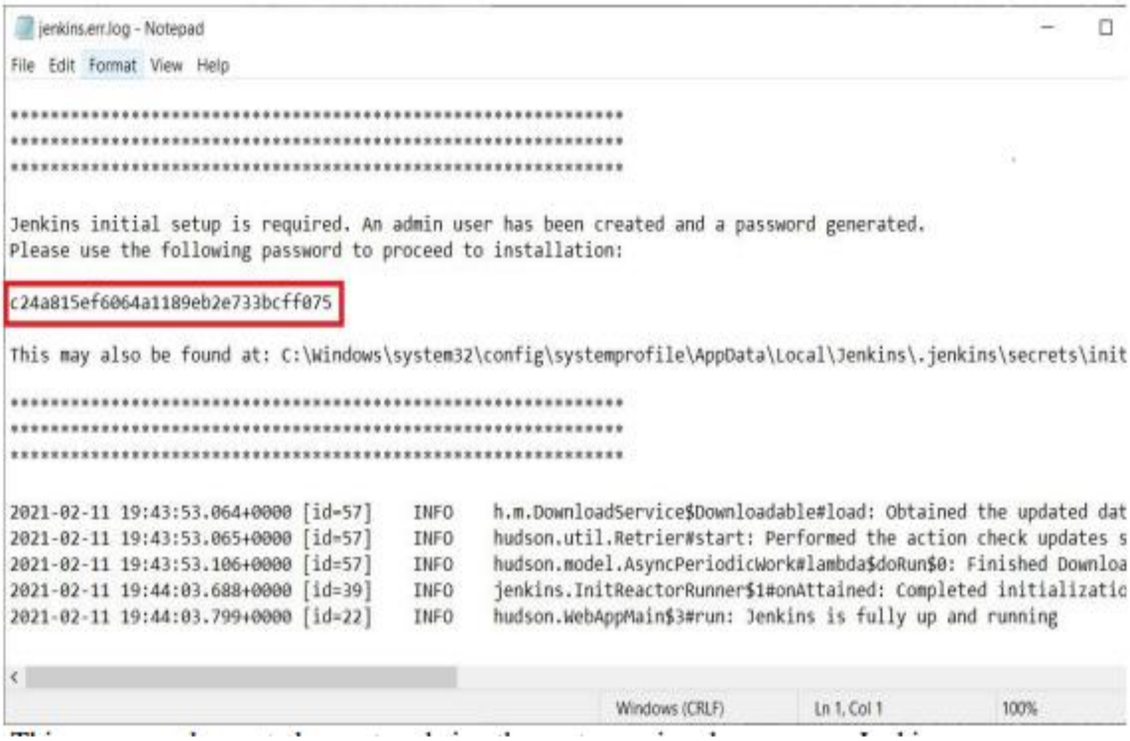
Step 3 Open the highlighted file and copy the content of the initialAdminPassword file



```
c5b286457bea4a038002ca957b5b5686
```

Step 4 On the Unlock Jenkins page,
paste this password into the Administrator password field and click Continue. Notes:

- You can also access Jenkins logs in the jenkins.err.log file in your Jenkins directory specified during the installation.
- The Jenkins log file is another location (in the Jenkins home directory) where the initial password can also be obtained.



```
*************************************************************
*************************************************************
*************************************************************

Jenkins initial setup is required. An admin user has been created and a password generated.
Please use the following password to proceed to installation:

c24a815ef6064a1189eb2e733bcff075

This may also be found at: C:\Windows\system32\config\systemprofile\AppData\Local\Jenkins\.jenkins\secrets\init

*************************************************************
*************************************************************
*************************************************************

2021-02-11 19:43:53.064+0000 [id=57]    INFO    h.m.DownloadService$Downloadable#load: Obtained the updated dat
2021-02-11 19:43:53.065+0000 [id=57]    INFO    hudson.util.Retrier#start: Performed the action check updates s
2021-02-11 19:43:53.106+0000 [id=57]    INFO    hudson.model.AsyncPeriodicWork#lambda$doRun$0: Finished Downloa
2021-02-11 19:44:03.688+0000 [id=39]    INFO    jenkins.InitReactorRunner$1#onAttained: Completed initializatic
2021-02-11 19:44:03.799+0000 [id=22]    INFO    hudson.WebAppMain$3#run: Jenkins is fully up and running
```

This password must be entered in the setup wizard on new Jenkins installations before you can access Jenkins's main UI. This password also serves as the default administrator account's password (with username "admin") if you happen to skip the subsequent user-creation step in the setup wizard.

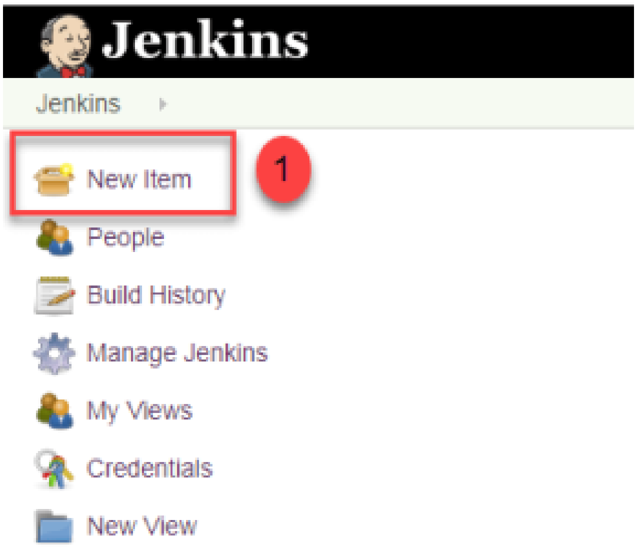
16.Create a container image for Hello world project

Create a container image for Hello world project And Setup build for container image using Jenkins (Hello world application)
How to Create a New Build Job in Jenkins

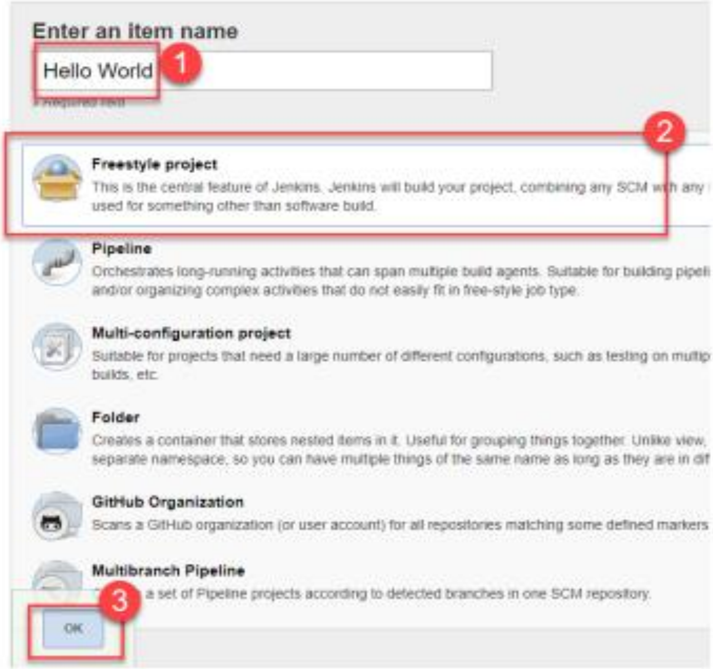
The freestyle build job is a highly flexible and easy-to-use option. You can use it for any type of project; it is easy to set up, and many of its options appear in other build jobs. Below is a step by step process to create job in Jenkin.

Step 1) Login to Jenkins To create a Jenkins freestyle job, log on to your Jenkins dashboard by visiting your Jenkins installation path. Usually, it will be hosted on localhost at http://localhost:8080 If you have installed Jenkins in another path, use the appropriate URL to access your dashboard as shown in the below Jenkins job creation example.

Step 2) Create New Item Click on "New Item" at the top left-hand side of your dashboard.



Step 3) Enter Item details In the next screen,
1. Enter the name of the item you want to create. We shall use the "Hello world" for this demo.
2. Select Freestyle project
 3. Click Okay



Enter an item name

Hello World

» Required field

Freestyle project
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any
used for something other than software build.

Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipeli
and/or organizing complex activities that do not easily fit in free-style job type.

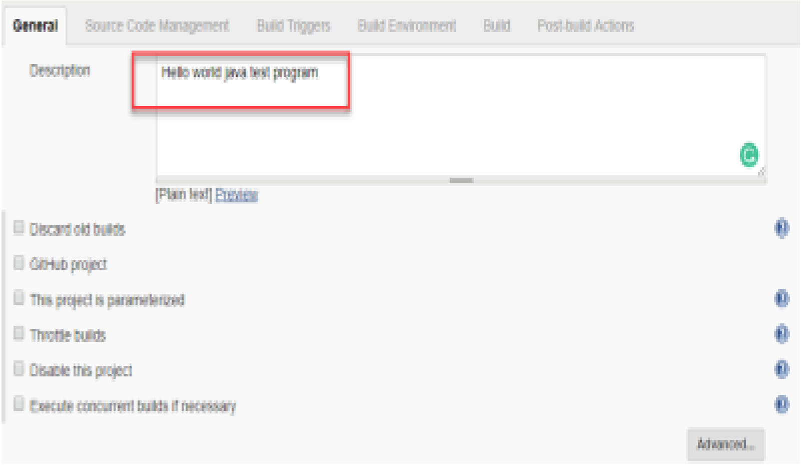
Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multip
builds, etc.

Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view,
separate namespace, so you can have multiple things of the same name as long as they are in dif

GitHub Organization
Scans a GitHub organization (or user account) for all repositories matching some defined markers

Multibranch Pipeline
a set of Pipeline projects according to detected branches in one SCM repository.

Step 4) Enter Project details
Enter the details of the project you want to test.

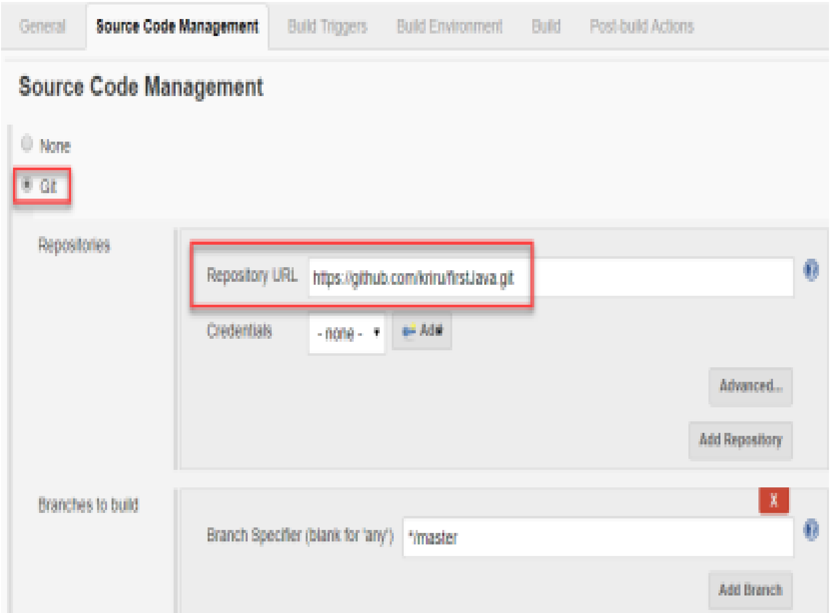


Description: Hello world java test program

[Plain text] Preview

□ Discard old builds
□ GitHub project
□ This project is parameterized
□ Throttle builds
□ Disable this project
□ Execute concurrent builds if necessary

Step 5) Enter repository URL
Under Source Code Management, Enter your repository URL. We have a test repository located at https://github.com/kriru/firstJava.git



Source Code Management

○ None
● Git

Repositories

Repository URL https://github.com/kriru/firstJava.git

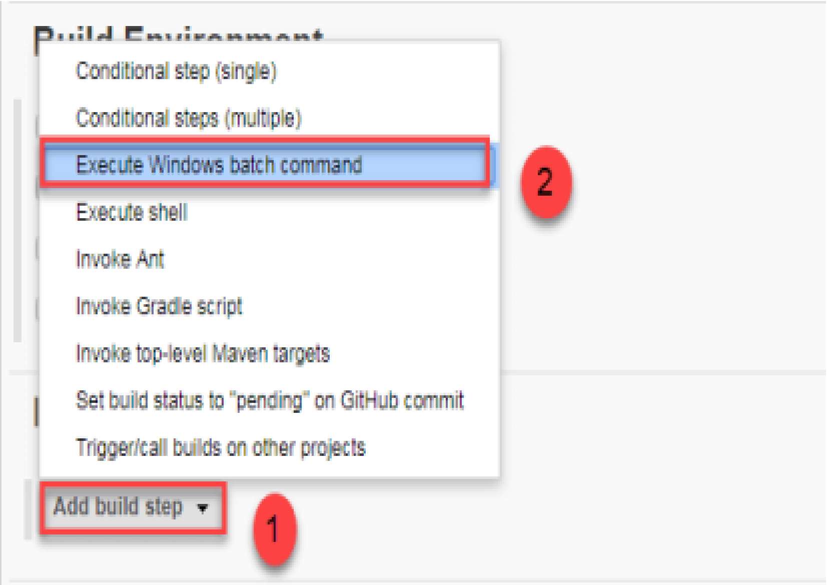
Credentials - none -

Branches to build

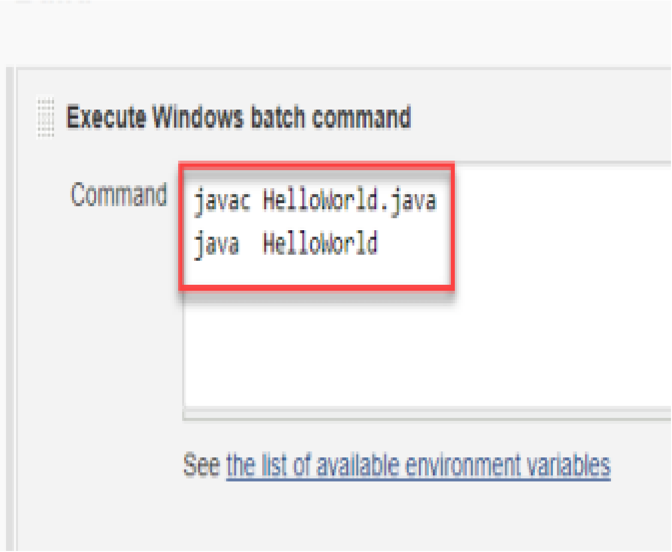
Branch Specifier (blank for 'any') */master

It is also possible for you to use a local repository
If your GitHub repository is private, Jenkins will first validate your login credentials with GitHub and only then pull the source code from your GitHub repository
Step 6) Tweak the settings Now that you have provided all the details, it’s time to build the code. Tweak the settings under the build section to build the code at the time you want.

You can even schedule the build to happen periodically, at set times. Under build,
1. Click on “Add build step”
2. Click on “Execute Windows batch command” and add the commands you want to execute during the build process.


Build Environment
Conditional step (single)
Conditional steps (multiple)
Execute Windows batch command
Execute shell
Invoke Ant
Invoke Gradle script
Invoke top-level Maven targets
Set build status to "pending" on GitHub commit
Trigger/call builds on other projects
Add build step
1
2

Here, I have added the java commands to compile the java code
I have added the following windows commands:
javac HelloWorld.java
java HelloWorld


Execute Windows batch command
Command
See the list of available environment variables

```
javac HelloWorld.java
java  HelloWorld
```

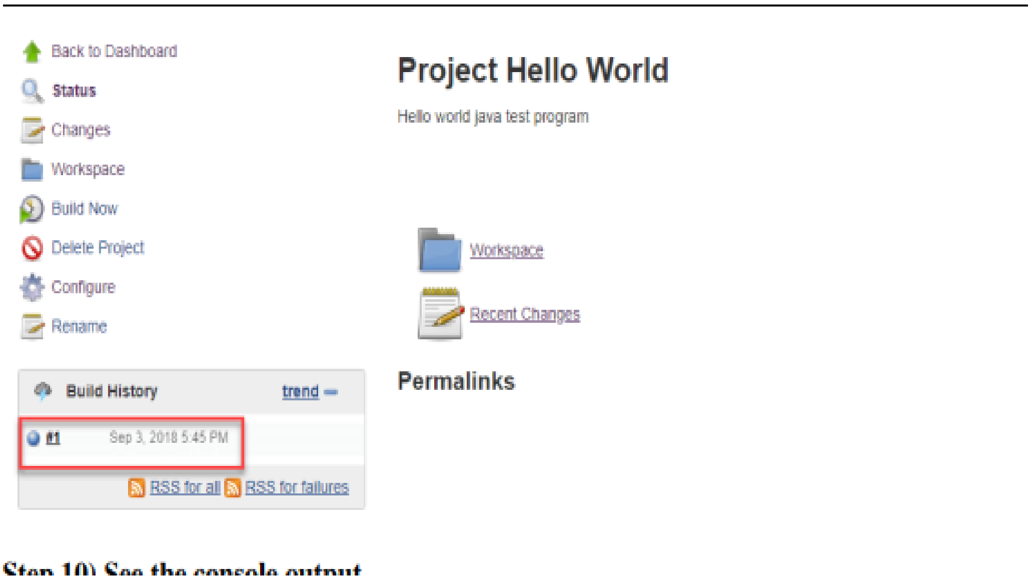
Step 7) Save the project When you have entered all the data,

1. Click Apply
2. Save the project.

Step 8:Build Source code Now, in the main screen, Click the Build Now button on the left-hand side to build the source code

Step 9) Check the status

After clicking on Build now, you can see the status of the build you run under Build History.



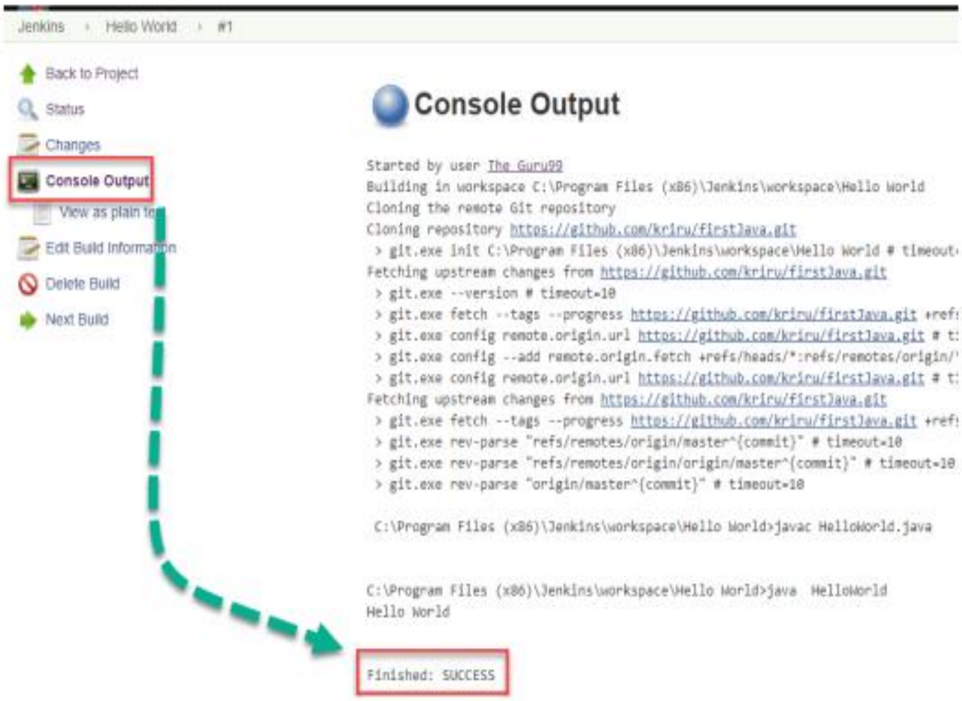
Project Hello World

Hello world java test program

#1 Sep 3, 2018 5:45 PM

Step 10) See the console output

Click on the build number and then Click on console output to see the status of the build you run. It should show you a success message, provided you have followed the setup properly as shown in the below Jenkins create new job example.



```
Started by user The Guru99
Building in workspace C:\Program Files (x86)\Jenkins\workspace\Hello World
Cloning the remote Git repository
Cloning repository https://github.com/kriru/firstJava.git
 > git.exe init C:\Program Files (x86)\Jenkins\workspace\Hello World # timeout
Fetching upstream changes from https://github.com/kriru/firstJava.git
 > git.exe --version # timeout=10
 > git.exe fetch --tags --progress https://github.com/kriru/firstJava.git +ref:
 > git.exe config remote.origin.url https://github.com/kriru/firstJava.git # t:
 > git.exe config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/
 > git.exe config remote.origin.url https://github.com/kriru/firstJava.git # t:
Fetching upstream changes from https://github.com/kriru/firstJava.git
 > git.exe fetch --tags --progress https://github.com/kriru/firstJava.git +ref:
 > git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10
 > git.exe rev-parse "refs/remotes/origin/origin/master^{commit}" # timeout=10
 > git.exe rev-parse "origin/master^{commit}" # timeout=10

 C:\Program Files (x86)\Jenkins\workspace\Hello World>javac HelloWorld.java


C:\Program Files (x86)\Jenkins\workspace\Hello World>java  HelloWorld
Hello World

Finished: SUCCESS
```

In sum, we have executed a HelloWorld program hosted on GitHub. Jenkin pulls the code from the remote repository and builds continuously at a frequency you define.

17.Organize Roleplay to understand the roles and responsibilities of QA and QC team.

Roles and Responsibilities of Quality Assurance(QA) Strategic roles and responsibilities of a Quality Assurance Manager In addition to their day-to-day duties as quality assurance managers, they may be asked to assume other strategic positions such as: • Managing the overall performance of the quality department

• Ensuring compliance with government regulations;
• Support the development of new products;
• Developing policies for the quality management system;
• Providing technical support to customers;
• Serving on committees responsible for developing and implementing strategies;
• Evaluating the effectiveness of existing programs; • Monitoring changes in market conditions;
• Preparing reports about current trends and future plans; • Conducting research into emerging technologies;
• Working with outside consultants; and
• Help develop marketing strategies designed to increase sales.
• Other tasks assigned by senior executives. Typical Skills of a Quality Assurance Manager
• The following skills may be found helpful in this position:
• Communication: The ability to communicate effectively both orally and in writing.
• Organization: A good sense of orderliness and time management.
• Problem-solving – An aptitude for analyzing situations and formulating solutions.
• Teamwork – Ability to get along well with others.
• Leadership – Leadership qualities such as initiative, self-confidence, assertiveness, decisiveness, and diplomacy.
• Creativity – Creative thinking abilities including imagination, originality, resourcefulness, flexibility, adaptability, and inventiveness.
• Judgment – Judgmental capabilities include discrimination, discernment, foresight, objectivity, and sound judgment.

• Analytical Thinking – The ability to think logically and critically about issues and solve problems. • Inquiry – Curiosity and interest in learning new things.
• Decision-Making – Decision-making skills that involve choosing alternatives and evaluating their relative merits.
• Planning – Planning skills that enable one to anticipate future needs and devise appropriate courses of action
. • Time Management – Time management skills that allow you to organize your activities efficiently and keep on schedule.

• Self Control – Self-control refers to the capacity to delay gratification and resist impulses.
• Adaptability – Adaptability involves the ability to adjust behaviour to suit changing circumstances. • Attention To Detail – Attention to detail can mean paying close attention to small details while performing routine tasks.
• Technical Skills - Technical skills and experience related to the industry.

Roles and Responsibilities of Quality Control(QC)

Responsibilities for Quality Control Inspector

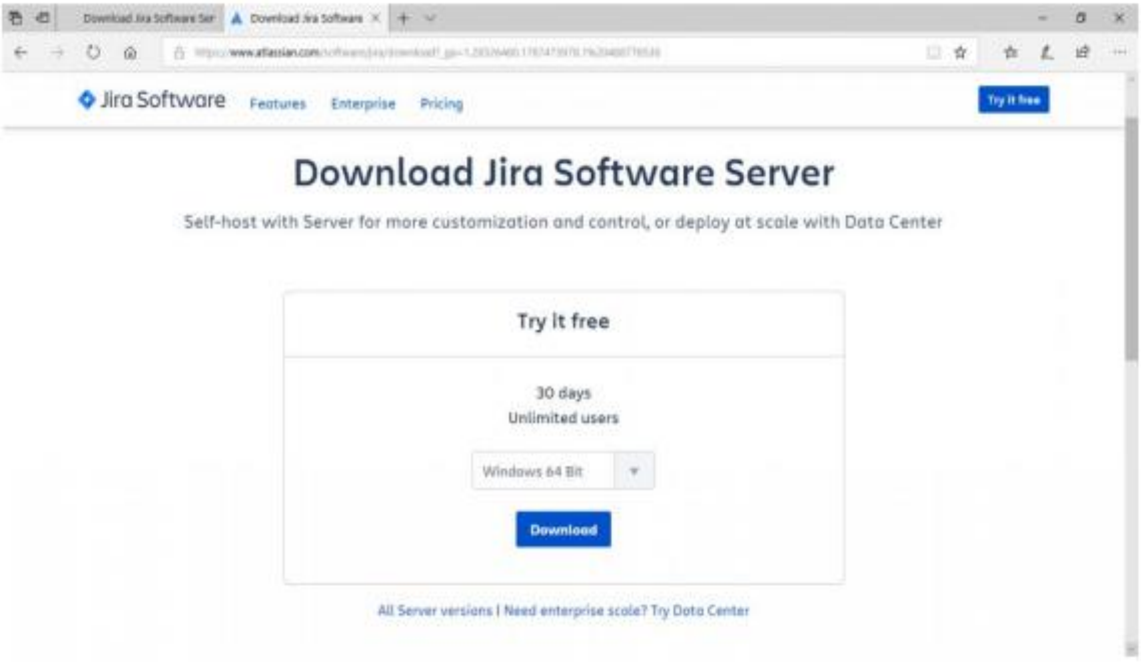
- Inspect products to ensure that they meet quality standards
- Create tests for quality control of products
- Disassemble product parts to inspect them individually
- Monitor production operations to ensure conformance to company specifications
- Direct assembly adjustments to ensure operations reflect quality standards
- Ensure products meet customer expectations based on company objectives
- Communicate the results of inspections and put forward corrective suggestions
- Write reports to document deficiencies and errors of products
- Carry out quality assessment measures of all the products ready to be shipped and incoming raw materials
- Take a thorough look at the plans, specifications, and blueprints to understand the product requirements
- Reject all the incoming raw materials fail to meet quality expectations and report the issue to the concerned department at the earliest
- Resolving quality-related issues adhering to deadlines
- Providing training to the quality assurance team

- Design an efficient design protocol which can be used across all domain
- Prepare documentation of the inspection process, which includes detailed reports and performance records
- Recommend improvement measures to the production process to ensure quality control standards are met
- Guide the production team about the quality control issues to enhance the quality of the product
- Monitor customer satisfaction levels
- Monitor the production phase at various levels\

18.create a jira similar tool.

Step 1 – To download and install Jira visit the official website of Atlassian. The link to the website ishttps://www.atlassian.com/software/jira/download?_ga=1.28526460.178747397 8.1 488778536

Step 2 – After selecting the type of Operating System in which you want to install Jira, look for the Download option and click on it. You can change the operating system type by clicking on the dropdown.



Download Jira Software Server

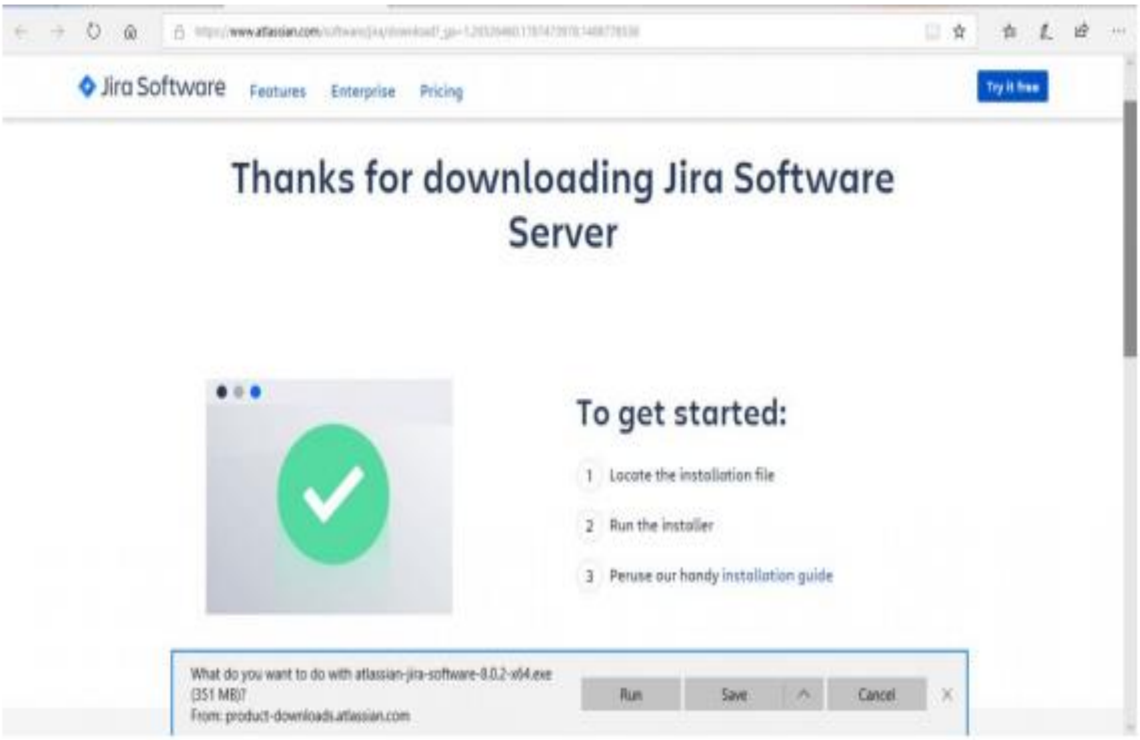
Self-host with Server for more customization and control, or deploy at scale with Data Center

Try it free

30 days
Unlimited users

Windows 64 Bit
Download
All Server versions | Need enterprise scale? Try Data Center

Step 3 – Once Jira is downloaded, click on the .exe file. After this, you will see that the Run confirmation pop-up is displayed, click on RUN to proceed. You can refer to the screenshot below.



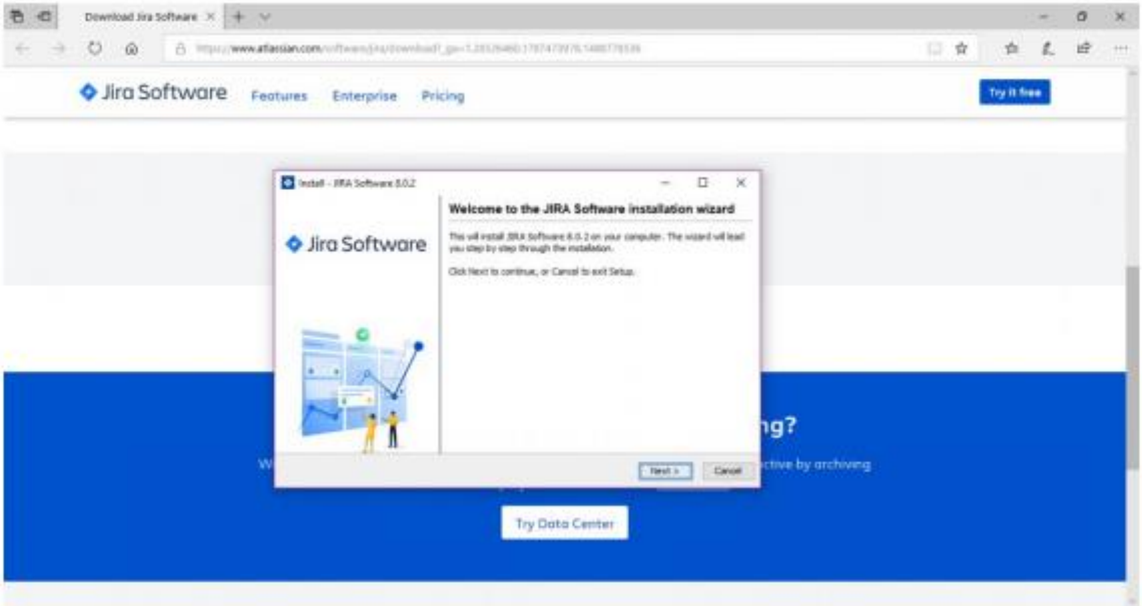
Thanks for downloading Jira Software Server

To get started:

1. Locate the installation file
2. Run the installer
3. Peruse our handy installation guide

What do you want to do with atlassian-jira-software-8.0.2-x64.exe (351 MB)?
From: product-downloads.atlassian.com

Step 4 – Notice that the JIRA installation wizard would be displayed. If so, clickNext



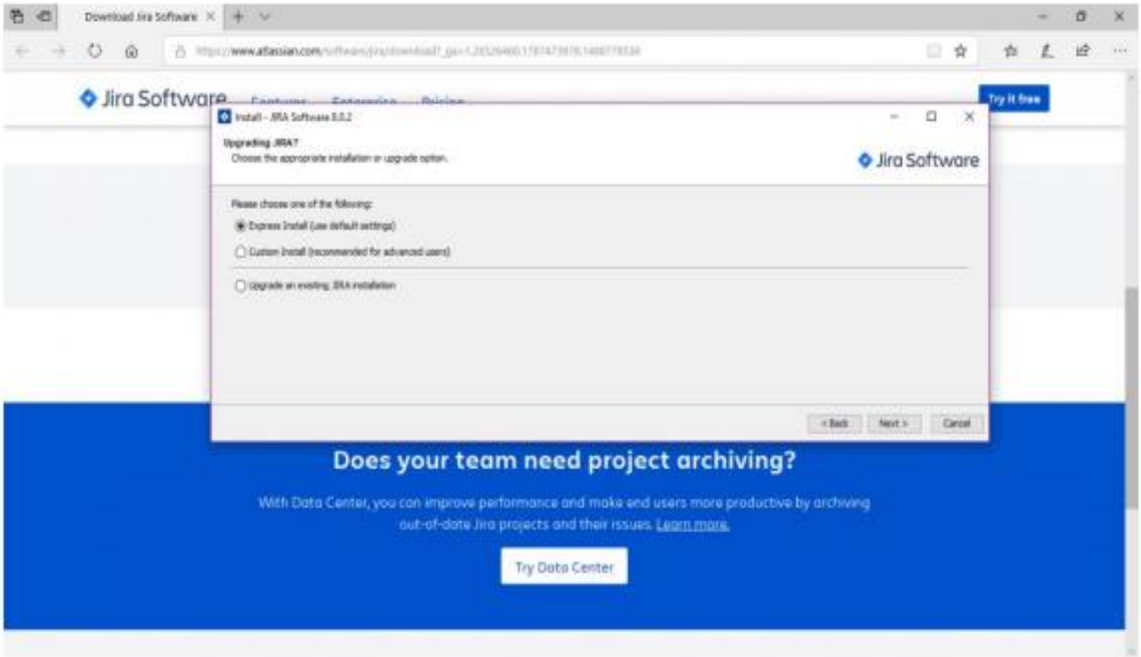
Welcome to the JIRA Software installation wizard

This will install JIRA Software 8.0.2 on your computer. The wizard will lead you step by step through the installation.

Click Next to continue, or Cancel to exit Setup.

Step 5 – Choose the desired installation option and then click on Next again. The installation summary would be displayed with the Destination Directory, Home Directory,
RMI Port, HTTP Port etc.

The screenshots for the same are attached below for your reference.



Install - JIRA Software 8.0.2

Upgrading JIRA?

Choose the appropriate installation or upgrade option.

Please choose one of the following:

● Express Install (use default settings)

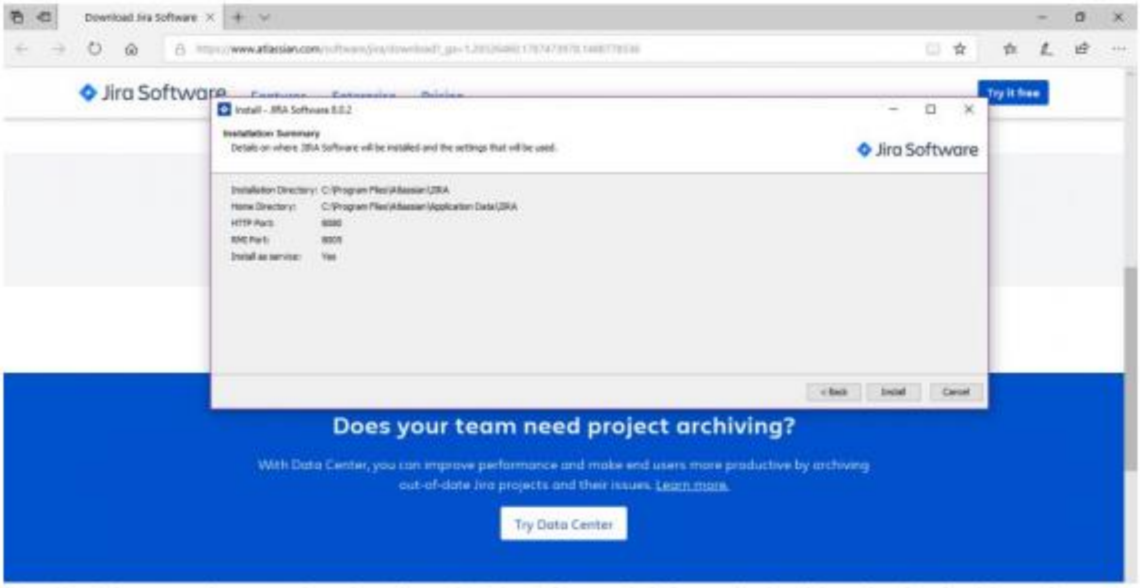
○ Custom Install (recommended for advanced users)

○ Upgrade an existing JIRA installation

Does your team need project archiving?

With Data Center, you can improve performance and make end users more productive by archiving out-of-date Jira projects and their issues. Learn more.

Step 6 – Click on Install. JIRA will start installing. It would take a few minutes for the installation to finish.



Install - JIRA Software 8.0.2

Installation Summary

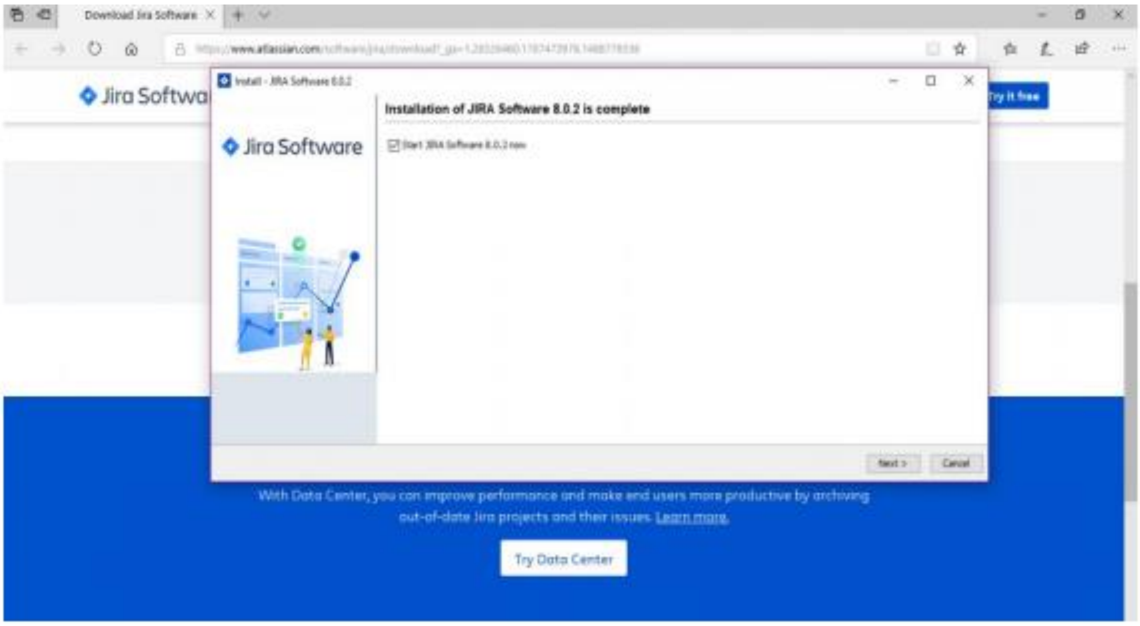
Details on where JIRA Software will be installed and the settings that will be used.

Installation Directory:	C:\Program Files\Atlassian\JIRA
Home Directory:	C:\Program Files\Atlassian\Application Data\JIRA
HTTP Port:	8080
RMI Port:	8005
Install as service:	Yes

Does your team need project archiving?

With Data Center, you can improve performance and make end users more productive by archiving out-of-date Jira projects and their issues. Learn more.

Step 7 – Please make sure that the "Start JIRA Software 8.0.2 now" checkbox is checked in order to start Jira automatically. After that click on Next, if not, it can be accessed using the Windows Start Menu shortcut



Install - JIRA Software 8.0.2

Installation of JIRA Software 8.0.2 is complete

☑ Start JIRA Software 8.0.2 now

With Data Center, you can improve performance and make end users more productive by archiving out-of-date Jira projects and their issues. Learn more.

Step 8 – Click the Finish button