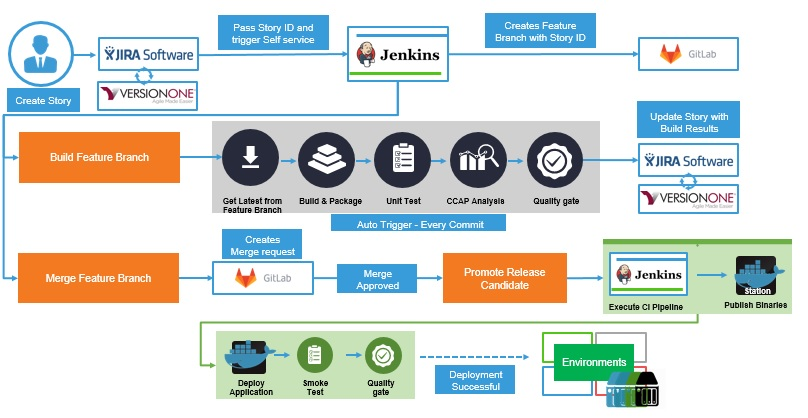
**Objective: Bot driven self-service automation for build & deployment.**

**Self-Service Illustration Architecture**



**Use Case: 1**

1. **Creating User story in Version One.**



Integration Points: virtual assist with Project Management Tool

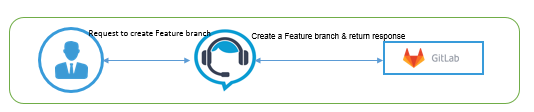
Actors: End user, Project Management Tool, Virtual Assist

Entity: Story Name

Endpoint: Jira rest api to create story

              Workflow:

1. End user login to virtual assist tool.
2. Virtual assist tool prompt the user on what action he/she need to perform.
3. End user request the virtual assist tool to create a new story in a sprint.
4. Virtual assist tool validate the logged in users access on Project Management Tool and create/reject the request as per his/her privilege.
5. **Creating feature branch in SCM.**



Integration Points: Virtual assist with SCM

Actors: End user, SCM, Virtual Assist

Entity: Branch name (or) Story Id

Endpoint: SCM api to create a branch

Workflow:

1. Virtual Assist prompt the user if he/she need to create a feature branch for the story created.
2. End user responds and Virtual Assist trigger a Jenkins job that create a feature branch with story ID in SCM tool.
3. **Creating feature branch jobs in Jenkins.**



Integration Points: Virtual assist with Jenkins

Actors: End user, Jenkins, Virtual Assist

Entity: Branch name (or) Story Id

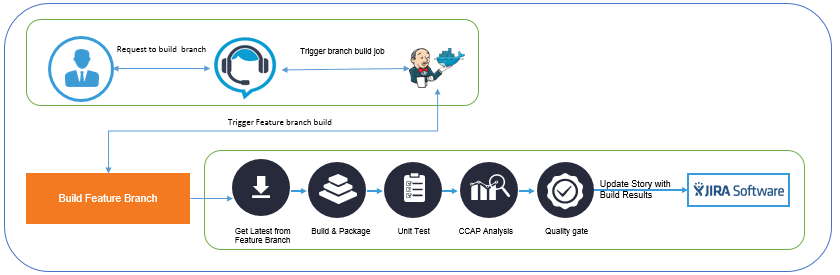
Endpoint: Jenkins rest api to trigger the template job.

          Workflow:

1. Virtual Assist prompt the user if he/she need to create a new feature branch job in Jenkins.
2. End user responds and Virtual Assist trigger Jenkins Admin job to create feature branch jobs.

**Use Case: 2**

1. **Triggering Feature branch build in Jenkins**



Integration Points: Virtual assist with Jenkins

Actors: End User, Jenkins, Virtual Assist

Entity: Branch name (or) Story Id

Endpoint: Jenkins rest api to trigger the branch job.

Workflow:

1. Virtual Assist prompt the user if he/she need to trigger the feature branch job in Jenkins.
2. End user responds and Virtual assist trigger Jenkins Feature job to build, package, test, on a dynamic slave docker container to perform the build and publish code quality result into SonarQube dashboard.

**2. Request for creating merge request.**



Integration Points: Virtual assist with SCM tool

Actors: End User, SCM, Virtual Assist

Entity: Branch name (or) Story Id

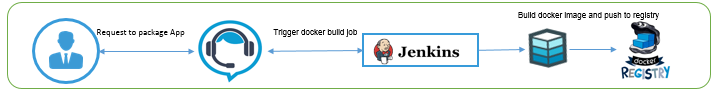
Endpoint: SCM rest api to create merge request.

Workflow:

1. Virtual Assist prompt the user whether the feature branch be merged to dev branch.
2. End user responds and Virtual assist connects to SCM and creates the merge request.

**Use Case: 3**

1. **Request to bake application with Docker image**



Integration Points: Virtual assist with Jenkins & Jenkins with Docker

Actors: Developer, SCM, Jenkins, Docker, Virtual Assist

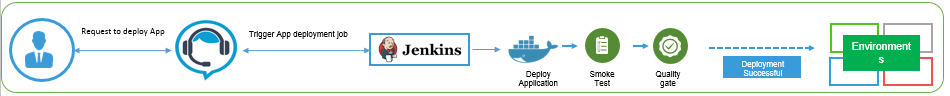
Entity: Branch name

Endpoint: Jenkins api to trigger the package & publish job.

Workflow:

1. Virtual assist prompt the user if the application need to be baked to a Docker image.
2. End user responds and Virtual assist request Jenkins to checkout the application from the merged branch, bake the Docker image and publish the image to Dev Snapshot repo.

**2. Request to deploy the Docker application.**



Integration Points: Virtual assist with Jenkins & Docker

Actors: Developer, SCM, Jenkins, Docker, Virtual Assist

Entity: Docker image id (or) name

Endpoint: Jenkins api to trigger the deploy job.

Workflow:

1. Virtual assist prompts if the application image need to be deployed from the registry.
2. End user responds and Virtual assist triggers the deployment through Jenkins and perform smoke test and publish the image into Dev pass repo.