Sprint 2 Stories

Detail: When tickets are created, fulfillers need UI Actions that they can use to progress a ticket from Open to Closed. Create the following UI Actions. We are also creating business rules that will automate some of the basic ticketing processes.

1. **HR UI Actions**
    a. Work in Progress
        - Visible when creating a new record and on existing records.
        - Visible when state is Open, On-hold and Resolved.
        - When the button is clicked, set the State to Work in Progress.
        - Order = 100.
        - Make this a Form button.

    b. On-hold
        - Visible when creating a new record and on existing records.
        - Visible when state is Open and Work in Progress.
        - When the button is clicked, set the State to On-hold.
        - Order = 110.
        - Make this a Form button.

    c. Resolve
        - Visible when creating a new record and on existing records.
        - Visible when state is Open, Work in Progress or On-hold.
        - When the button is clicked do the following:
            - Set the State field to Resolved using client-side script. This is needed to expose the Closure Information and enforce UI policies tied to the fields on the Closure Information tab.
            - Set the Comments field on the Notes tab Mandatory. Do not do this as a UI policy, write this as a client-side script on the UI action.
            - Create a function that will run your server-side script. The script should simply update the record.
        - Order = 120
        - Make this a Form button.

    d. Close
        - Visible on existing records only.
        - Visible state is Resolved.
        - When the button is clicked do the following:
            - Set the State field to Closed using client-side script. This is needed to expose the Closure Information and enforce UI policies tied to the fields on the Closure Information tab.
            - Set the Comments field on the Notes tab Mandatory. Do not do this as a UI policy, write this as a client-side script on the UI action.

- Create a function that will run your server-side script. The script should simply update the record.

- Order = 130
- Make this a Form button.

e. Cancel
- Visible on existing records only.
- Visible when the ticket is active.
- When the button is clicked do the following
  - Using client-side script set the Comments field to Mandatory.
  - Set the State to Canceled using a server-side script and update the record.
- Order = 140.
- Make this a Form button.

Note that all UI actions should redirect to the current page which would be the HR ticket form.

2. HR Task UI Actions

a. Work in Progress
- Visible when creating a new record and on existing records.
- Visible when state is Open, On-hold and ~~Resolved~~.
- When the button is clicked, set the State to Work in Progress.
- Order = 100.
- Make this a Form button.

b. On-hold
- Visible when creating a new record and on existing records.
- Visible when state is Open and Work in Progress.
- When the button is clicked, set the State to On-hold.
- Order = 110.
- Make this a Form button.

c. Close
- Visible on existing records only.
- Visible active is true.
- When the button is clicked do the following:
  - Set the State field to Closed using client-side script. This is needed to expose the Closure Information and enforce UI policies tied to the fields on the Closure Information tab.
  - Set the Comments field on the Notes tab Mandatory. Do not do this as a UI policy, write this as a client-side script on the UI action.
  - Create a function that will run your server-side script. The script should simply update the record.

- Order = 120
- Make this a Form button.

3. HR Business Rules

a. Create a business rule called State is Resolved that will do the following.
    - Run before the record is updated.
    - Run when records are inserted or updated.
    - Run when state changes to Resolved.
    - Run a script that will do the following:
        - Search all existing HR Tasks associated with the HR ticket that are active.
        - If there are active tasks abort the action and show an error message "Please close all active tasks before Resolving this ticket".
        - If there are no active tasks set t he following fields
            a. Resolved by – current logged in user. Use the appropriate GlideSystem User Object for this.
            b. Resolved – current date/time. Look into GlideDateTime to do this.

b. Create a business rule called Clear Closure fields that will do the following.
    - Run before the record is updated.
    - Run when records are inserted or updated.
    - Run when state changes and when state is one of the following (Open, Work in Progress, On-hold)
    - Run a script that will clear the following fields:
        - Close Code
        - Close Notes
        - Closed by
        - Resolved
        - Resolved by

c. Create a business rule called Set State to Work in Progress
    - Run before the record is updated.
    - Run when records are inserted or updated.
    - Run when Assigned to changes and Assigned to is not empty and State is Open.
    - Set the state to Work in Progress.

d. Create a business rule called Copy Comments to Task Work note.
    - Run before the record is updated.
    - Run when records are inserted or updated.
    - Run when Additional comments changes.

- Run a script that will do the following:
  - Search for all active HT tasks associated with the HR Ticket.
  - Iterate through the list and update each task work note with the comments from the HR Ticket.