# Revisions

|  |  |
| --- | --- |
| MOCKUPS FILE | WHAT’S NEW |
| AgentUI-WIP-1.zip | Self-contained Agent UI - brainstorm |
| Agent-UI-WIP-2.zip | Transition from UI brainstorm to realistic first iteration of Agent UI:   * Temporarily remove idea of auth for Agent UI. * Remove any appearance of branding. * Remove idea of ability to test config settings before saving them. * Remove idea of Agent storing its activation key. * Add Agent knowledge of the VX System Name, Type, and Version to which it is connected. * Add Agent’s knowledge of its own version. * Add some validation for activation key and VX password entered.   Changed HTML files:   * activation.html * remove/hide branding img and creation of admin user * add agent version at bottom * add validation that key can be successfully parsed * config.html * remove/hide Log Out button in header * minor changes to cloud (now Pelco Breeze) connection fields, including remove/hide Port setting and adding VX System Name, Type, Version * add agent version at bottom * editActivationKeyModal.html * change modal title and body text * add validation that key can be successfully parsed * remove/hide Test Key parts * editVxSettingsModal.html * change how validation that Password is not blank works * remove/hide editability of Port   Changed LESS files (can probably just replace existing files with new ones):   * base.less * change header background image to solid color * add class .agent-version at end * buttons.less * add class .button-muted to .button-row-centered, .button-row-left, and .button-row-right * modules/activation.less * almost the whole file changed, due to removing branding, removing creation of admin user, enlarging textarea input and adding validation of key format * modules/editActivationKeyModal.less * almost whole file changed, due to enlarging textarea input, adding validation of key format, and remove/hide Test Key * modules/editVxSettingsModal.less * remove/hide Test Settings |

# Workflows

## User activates agent for first time

1. When agent installation completes, user is prompted (with link) to go to agent UI.
2. If no activation key has been entered for this agent, show prompt to enter a key (activation.html).
3. User enters activation key for the agent.
4. User clicks “Activate Agent” button.
5. UI sends input value to agent, agent validates that the key can be parsed.
6. Agent uses values parsed from key to attempt to connect to cloud.
7. UI shows the config.html page, showing result of Agent’s attempt to connect to cloud in “Pelco Breeze Connection” section of config.html.
8. UI indicates in “VX Connection” section of config.html page that a password must be set for the “cloud\_agent” user in VX.
9. User clicks on the “edit” icon in the “password” field.
10. UI shows editVxSettingsModal.html. All inputs show current values. Agent should ship with default values, such as ipAddress = 127.0.0.1, pollInterval = 60 secs, but \*no\* default password. If a user clears one of the inputs with a default value, then on blur out of the input, it can take its default value again.
11. User enters password.
12. User clicks “Save” button.
13. Agent saves VX connection settings and attempts to connect to VX with new settings.
14. UI goes back to config.html page, with info refreshed. There is no longer an “edit” icon in the “password” field; bullet characters show that a password has been saved.

## User updates activation key

1. User goes to agent UI with browser.
2. If an activation key has already be entered for this agent UI shows config.html page.
3. User clicks “edit” icon in upper right of “Pelco Breeze Connection” section.
4. UI shows editActivationKeyModal.html.
5. User enters new activation key.
6. User clicks “Save” button.
7. UI validates that the activation key entered can be parsed.
8. Agent saves new settings and attempts to connect to cloud with them.
9. UI goes back to config.html page, with info refreshed.

## User updates VX connection settings

1. User goes to agent UI with browser.
2. If an activation key has already be entered for this agent UI shows config.html page.
3. User clicks “edit” icon in upper right of “VX Connection” section.
4. UI shows editVxSettingsModal.html
5. User edits desired fields.
6. User clicks “Save” button.
7. Agent saves new VX connection settings and attempts to connect to VX with them.
8. UI goes back to config.html page, with info refreshed.