## What is this?  
  
  
  
Rocket.Chat Apps can provide a starting point for bespoke chat ops / messaging automation, to replace the current Internal Hubot.  
  
  
  
Note the distinction between \_internal\_ and \_external\_ bots:  
  
  
  
- External hosted bots use a platform or framework adapter to login and operate a user account. They are suited to messaging-heavy interactions and/or where maintenance or processes are independent of Rocket.Chat.  
  
- Internal bots are tightly coupled with the architecture of Rocket.Chat, but (at least currently) have limited features for conversational UI. Their provisioning and operation is dependent on the Rocket.Chat server.  
  
  
  
## Why do it?  
  
  
  
The internal Hubot is outdated, it’s utility for enterprises and teams is limited and its documentation and usability is a challenge blocking many users from making the most of it.  
  
  
  
Rocket.Chat now has much better options for messaging automation from progress toward an app marketplace model and the recent updates towards an SDK for multiple bot platform/framework adapters.  
  
  
  
Integrating custom processes into the core codebase and the generic “Rocket.Cat” bot identity is harder to maintain and can be confusing for users to interact with.  
  
  
  
## How will it help?  
  
  
  
- Instances can maintain bot driven logic without requiring a fork of core Rocket.Chat  
  
- Customised identity for bots in a Rocket.Chat instance distinct from the “system bot”  
  
- Clearer semantics between system notifications and instance’s specific business logic  
  
- Access to apps API endpoints will open new features to internal bots  
  
- Multiple internal bots can provide different behaviours from unique bot user profiles  
  
  
  
## What does it need?  
  
  
  
- App with same features as a bot: receive, send, reply, store data, call API  
  
- Provide http listeners for triggering internal methods / interactions  
  
- Make http requests to get data to trigger or populate a message send  
  
- Documentation for getting started with defining a new bot app and writing scripts  
  
- Boilerplate examples to show simple common use cases achieved by bot app  
  
  
  
## What’s the plan?  
  
  
  
- Identify some use cases of current internal Hubot extensions  
  
- Build and document example implementations with Rocket.Chat Apps to replace Internal Hubot interactions  
  
- Build a replacement for Rocket.Cat as default packaged app-bot  
  
- Update docs for a clear purpose for the single system bot as a simple service butler, that will no longer be extended with instance specific scripts.  
  
  
  
## How can we help?  
  
  
  
If this spec effects you and you’d like to help out, what we need at this stage is to understand the types of use cases currently implemented with the Internal Hubot, so we can provide useful templates and guides to migrate your Hubot scripts into Rocket.Chat apps.  
  
  
  
Related to #10456, #6640, #8025