Pharo TechTalk: Discord Demo

Tuesday, Nov 21, 2017
5:00 PM - 7:00 PM (UTC+01:00, Paris)
01:00 PM - 03:00 PM (UTC-03:00, Chilean Time)
Pharo TechTalk: Discord Demo

Juraj Kubelka

https://github.com/JurajKubelka/DiscordSt

Metacello new
baseline: #DiscordSt;
repository: 'github://JurajKubelka/DiscordSt/src';
load.
TechTalk Outline

1. **Webhook** is a low-effort way to **write messages** to channels in Discord using a secret URL. GitHub is supported directly.

2. **Bot Application** is a “chatbot” that can **read and write messages** and manage servers and channels. E.g., Lighthouse by Kilon: “what is pharo?”

3. **Standard User** account with an email and password. Notice that using such account to connect bots is strictly prohibited.
1. **Webhook** is a low-effort way to **write messages** to channels in Discord using a secret URL. GitHub is supported directly.

2. **Bot Application** is a “chatbot” that can **read and write messages** and manage servers and channels. E.g., **Lighthouse by Kilon**: “what is pharo?”

3. **Standard User** account with an email and password. **Notice that using such account to connect bots is strictly prohibited.**
Webhook Configuration

Copy the Webhook URL
Webhook Examples

```ruby
DSWebhookClient new
  message: Hello from the [Pharo](http://pharo.org) TechTalk :-)';
post

DSWebhookClient new
  username: 'Pharo Screenshot';
  avatarUrl: 'http://files.pharo.org/media/logo/icon-opaque-512x512.png';
  fileName: 'screen.png' morph: World;
post
```
Webhook Examples: Embed

DSWebhookClient new
  username: 'Pharo Screenshot as Embed';
  avatarUrl: 'http://files.pharo.org/media/logo/icon-opaque-512x512.png';
  embedTitle: 'Discord Bot Announcement';
  embedDescription: 'Embed is a structured message. See [documentation](https://discordapp.com/developers/docs/resources/channel#embed-object)';
  embedUrl: 'https://github.com/JurajKubelka/DiscordSt';
  embedMorph: World;
  embedColor: Color green;
  embedFooterText: 'Discord API in Pharo';
  embedFooterIconUrl: 'https://discordapp.com/assets/2c21aeda16de354ba5334551a883b481.png';
  post
Webhook Use Case #1

Sharing Experience within Pharo

```
^ BlElement new
size: 300@200;
relocate: 100 @ 100;
geometry: (BlRectangle cornerRadius: 40);
background: self linearGradient;
yourself
```
Roassal Script of the Day

Script of the day: Bundle Edges examples
More information about Roassal on http://agilevisualization.com/

```plaintext
| b v |
v := RTView new.
b := RTBundleBuilder new.
view: v.
b explore: Collection using: #subclasses.
b bezier color: Color blue trans.
b useBezierLineWith: #dependentClasses.
b edgeBuilderDo: [ :ebuilder |
  ebuilder shape: (RTArrowedLine new line: b bezier; yourself).
  ebuilder connectToAll: #dependentClasses ].
```

---

**LIBRARIES**
- databases
- polymath
- roassal
- roassal-scriptoftheday
- seaside
- moose
- bloc
- ai
- ai-scriptoftheday

**INTERMITTENT**
- techtalk
- sprint
- events
- jobs
- gci-students
Webhook Use Case #2

Be informed about strange behavior on my server
TechTalk Outline

1. **Webhook** is a low-effort way to write messages to channels in Discord using a secret URL. GitHub is supported directly.

2. **Bot Application** is a “chatbot” that can read and write messages and manage servers and channels. E.g., Lighthouse by Kilon: “what is pharo?”

3. **Standard User** account with an email and password. Notice that using such account to connect bots is strictly prohibited.
Bot App Configuration

Create Bot App
https://discordapp.com/developers/applications/me

Generate Permissions and Authorize the Bot
https://discordapi.com/permissions.html

Link: https://discordapp.com/oauth2/authorize?client_id=INSERT_CLIENT_ID_HERE&scope=bot&permissions=0
Bot App Examples

```
DSBot new
token: 'BOT-TOKEN';
servers.
```

```
DSBot new
token: 'BOT-TOKEN';
with: [:bot | bot message
    content: 'Hi! I am a bot :-)';
    sendToServer: 'Pharo' channel: 'general' ].
```
Bot Application

How can I install Bloc?

Is there a documentation about Spec?

Do we have a library to parse JSON?
Bot Use Case: Expertise

```
bot := DSBot new.
bottoken: 'BOT-TOKEN'.
bottoken connect.

answer := DSExpertiseAnswer new bot: bot; yourself.

bot announcer
when: DSGatewayMessageAnnouncement
do: [:ann | DSExpertiseQuestion
message: ann message
answer: answer ].
```
Bot (Chatbot) Research

Should bots be treated as virtual team members or as tools? [Psychology, CSCW]

How to make interactions with the bot more effective and enjoyable? [HCI, IxD, Information Visualization]

What roles/activities should bots be used for? Where shouldn’t they be used? [SE, Organizational behaviour]

How to integrate bots into a team/organization? [CSCW, SE]

How autonomous should the bots be? How does that fit with task ownership? [CSCW]

How to improve bot effectiveness and productivity? [ML, AI, NLP]

Is it possible to automate some critically-relevant abilities that humans rely on? [Formal verification and synthesis, ML, Automatic Reasoning]

@alexeyzagalsky  March 15th, 2017

https://twitter.com/alexeyzagalsky/status/927661437350060032
1. **Webhook** is a low-effort way to **write messages** to channels in Discord using a secret URL. GitHub is supported directly.

2. **Bot Application** is a “chatbot” that can **read and write messages** and manage servers and channels. E.g., Lighthouse by Kilon: “what is pharo?”

3. **Standard User** account with an email and password. 
   *Notice that using such account to connect bots is strictly prohibited.*
Client Configuration
Client Examples

```ruby
client := DSClient new.
client
  email: 'email@example.com'
  password: 'password'.
client login.
client.
```

Servers

- User Guild: Discord API
  - User Guild: Pharo
    - Category: For new users
      - Text Channel: learning-help
      - Text Channel: mooc
    - Category: PHARO
      - Text Channel: general
      - Text Channel: development
    - Category: Libraries
      - Text Channel: offtopic
      - Text Channel: iceberg
      - Text Channel: bootstrap
    - Category: Libraries
      - Text Channel: databases
Client Use Case: Ask

Don’t you know how to solve your task? Ask!

"Require:
GT-Playground-JurajKubelka.151
GT-Tests-Playground-JurajKubelka.5"
Gofer it
smalltalkhubUser: 'Moose' project: 'GToolkit';
package: 'GT-Playground';
package: 'GT-Tests-Playground';
load.
Client Use Case: Answer

```plaintext
Hi!

How should I modify the mapping to be able to parse the following example?

rectangleJson := '{
  "origin": null,
  "corner": null
}';

(NeoJSONReader on: rectangleJson readStream)
mapInstVarsFor: Point;
for: Rectangle do: [ mapping |
  (mapping mapInstVar: #origin) valueSchema: Point.
  (mapping mapInstVar: #corner) valueSchema: Point.
  nextAs: Rectangle.
]
```

Download

Edit & Share
Discussion

What other use cases would you like to try?

What would you like to discuss directly from Pharo?