Component Diagram for Web App Biocycle

Legend:
- — — → (blue) Uses
- — — → (black) Uses [JSON/HTTPS]
- Controller (orange)
- Bean (green)
- Proxy (magenta)
- External resource (gray)

Web App Biocycle

Account creation Controller [Component: Java,SpringBoot Rest Controller]

Sign in Controller [Component: Java,SpringBoot Rest Controller]

Organisation Managment Controller [Component: Java,SpringBoot Rest Controller]

GiveAway Managment Controller [Component: Java,SpringBoot Rest Controller]

Collection Managment Controller [Component: Java,SpringBoot Rest Controller]

Collection Planning Controller [Component: Java,SpringBoot Rest Controller]

ProductBatch entry Controller [Component: Java,SpringBoot Rest Controller]

ProductBatch storage Controller [Component: Java,SpringBoot Rest Controller]

Redistribution Managment Controller [Component: Java,SpringBoot Rest Controller]

Staff Component [Component: Spring Bean]
Model Object

Organisation Component [Component: Spring Bean]
Model Object

GiveAway Component [Component: Spring Bean]
Model Object

CollectionRun Component [Component: Spring Bean]
Model Object

ProductBatch Component [Component: Spring Bean]
Model Object

Redistribution Component [Component: Spring Bean]
Model Object

CRUD staff Proxy [Component: Spring Proxy]
Provides functionalities related to account creation and sining in ...

CustomerManagment Service Proxy [Component: Spring Proxy]
Provides functionalities to approve / refuse partnership

GiveAway Service Proxy [Component: Spring Proxy]
Provides funtionalities to accept totally or partially / refuse giveaway

CollectionManagment Service Proxy [Component: Spring Proxy]
Provides funtionalities to see containers to be collected, to add containers to a collection lap, to see work planning for bike collector

Inventory Service Proxy [Component: Spring Proxy]
Provides funtionalities to entry the received batch into the system, retrieve batch storage

ProductDispatch Service Proxy [Component: Spring Proxy]
Provides funtionalities to manage offers intended for receivers

API Gateway / Resource server [Container MS: Java,SpringBoot, Zuul, OAuth2]
Root and Filter requests to Microservices