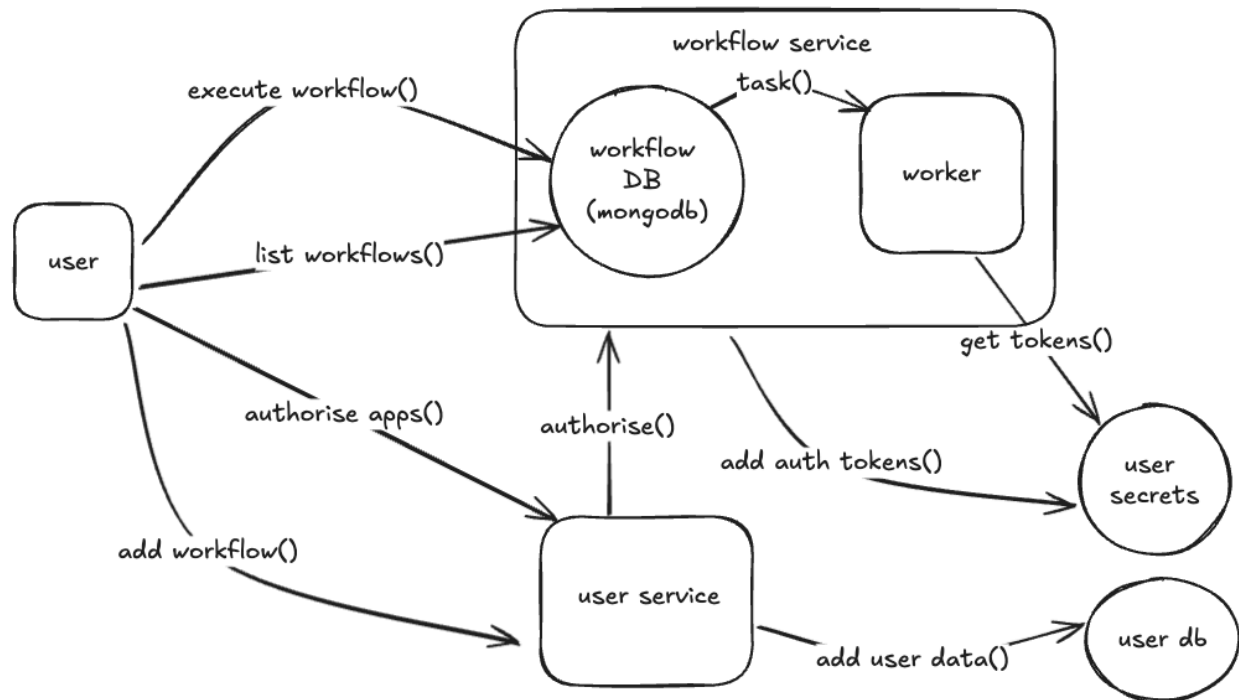


Brief: Workflow automation system similar to [Zapier](#). This system allows users to integrate automated workflows that connect various web applications and services.

Functionalities:

1. List workflows
2. Add/Create a workflow for a user
3. Execute a workflow

High Level Design:



Database Schema:

<Applications> (To store 3rd party API's)

Suite - company_name (Primary Index)

App - app_name (Secondary Index)

Action - action to perform

Api - API corresponding to the action

Example: (There is no database connected. These are stored in json/py files)

```
applications = {"google":
    {"gmail": {"read":
        "https://www.googleapis.com/auth/gmail.readonly"}},
```

```

        "google_drive": {"upload":
"https://www.googleapis.com/auth/drive.file"},
        },
    }

```

<Workflows> (Main db that contains all the workflows in our system)

Id - unique identifier (Primary Index)

Metadata:

Name - Workflow name

Source app - List of apps that represent the source in a workflow

Destination app - List of apps to save data from source apps

Steps: Workflow worker to perform the task of the workflow

Note: Source app/ Destination app: List of dicts with app details like app_name, suite, and actions to be done in this workflow

Example: (There is no database connected. These are stored in json/py files)

```

workflows = {"1":
    {"metadata": {"name": "gmail_gdrive",
                  "source_app": [{"app_name": "gmail",
                                  "suite": "google",
                                  "actions": ["read"]}],
                  "destination_app": [{"app_name": "google_drive",
                                       "suite": "google",
                                       "actions": ["upload"]}]},
    },
    "steps": GmailToDriveWorkflow
}

```

<Integrations> (To show active workflows in the system to a user)

Name - Workflow name

Source app - List of apps that represent the source in a workflow

Destination app - List of apps to save data from source apps

Description - To describe the purpose of this workflow

Status - To show if the workflow is active or inactive to add for new users.

Note: Source app/ Destination app: List of dicts with app details like app_name, suite, and actions to be done in this workflow

Example: (There is no database connected. These are stored in json/py files)

```

integrations = {
    "gmail_gdrive": {"source_app": [{"app_name": "gmail", "suite": "google"}]},

```

```

        "destination_app": [{"app_name": "google_drive", "suite":
"google"}],
        "description": "Upload attachments in mails to google
drive",
        "status": 1}
}

```

<Users> (store user related information)

Id - unique identifier (Primary Key)

Workflows - List of workflow id's user added to his account.

Example: (There is no database connected. These are stored in json file)

```

{
  "user1": {
    "workflows": [
      "1"
    ]
  }
}

```

Important considerations:

1. User Secret tokens are kept separate for each workflow. This is done to keep workflows of a user independent and give the ability to delete an entire workflow without any dependency on other workflows.
2. This step also improves the security as a token corresponding to a workflow only has permissions necessary to perform the tasks in that workflow. The scope of a token is robust.
3. So, the user has to authorize apps in the workflow only once after adding a workflow.

Future Improvements:

1. For more complex workflows with multiple integrations, the workflow can be stored as directed graphs with edges representing the data schema and the nodes being the task worker capable of handling the task.

Note: For now since this is a basic implementation of gmail to drive workflow which has just two app integrations, workflows is defined as source and destination apps.