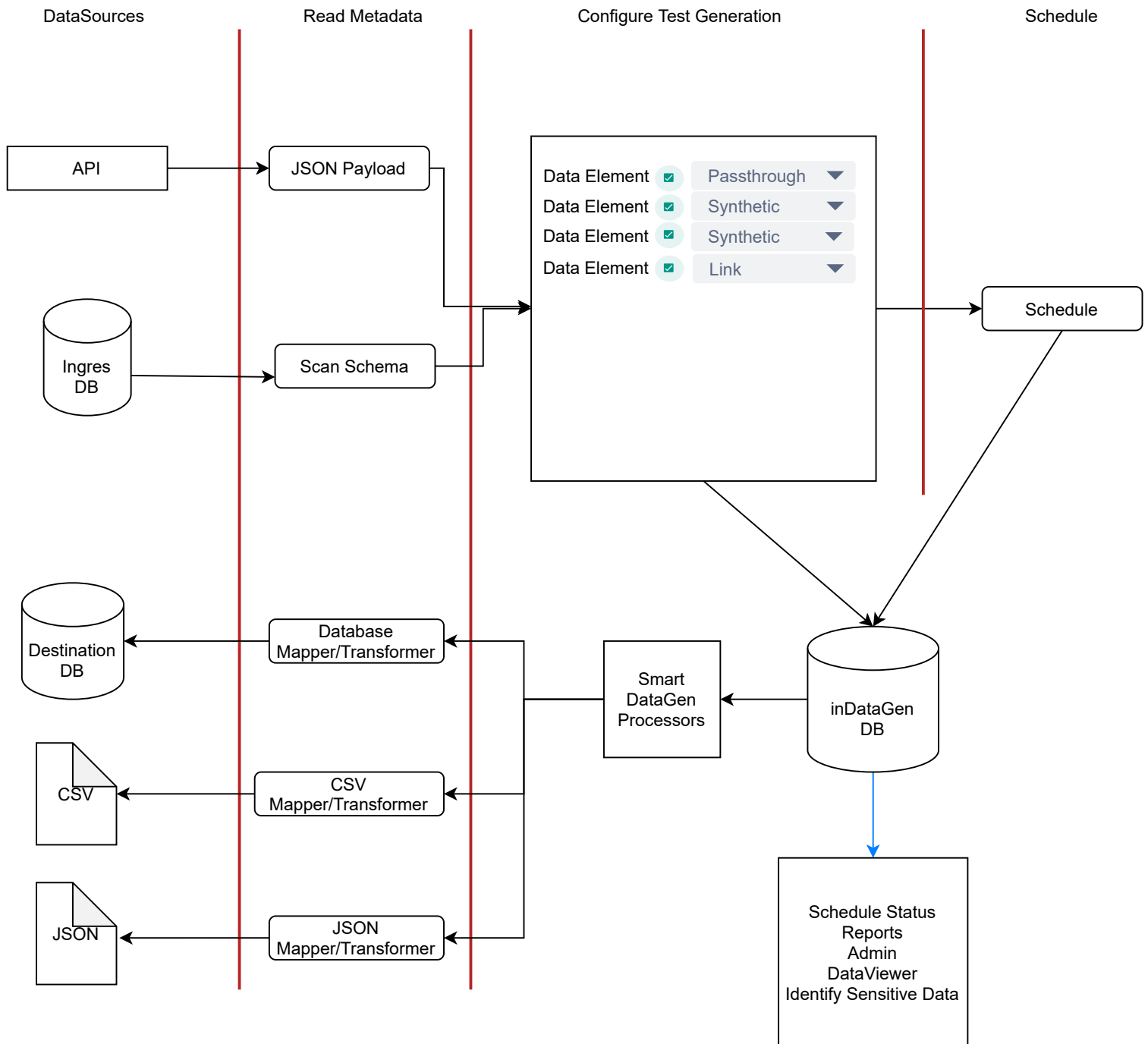
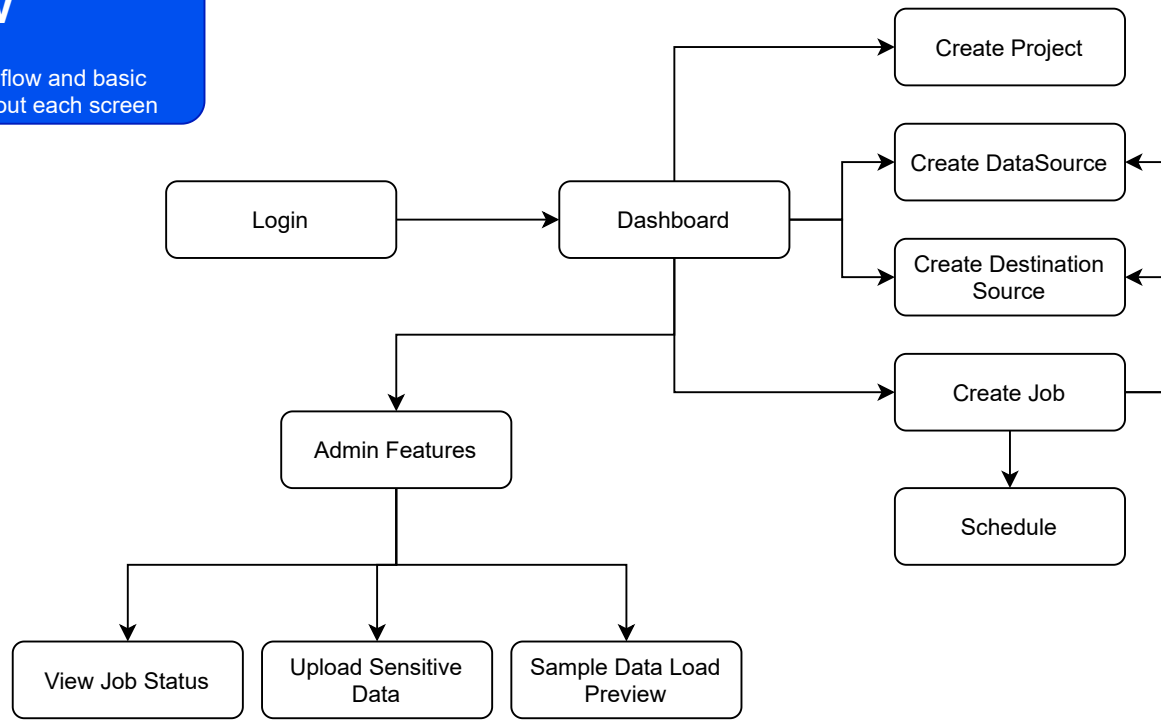


# Notional Architecture



# UI Flow

Defines the UI flow and basic information about each screen



## Create DataSource

1. Type of the datasource
2. Connection details
3. Verify Connectivity

## Create Destination Source

1. Type of the datasource
2. Connection details
3. Verify Connectivity

## Create Job

1. Select source connector
2. Select destination connector
3. Scan Schema
4. Provide tables to select
5. Provide columns to select
6. Provide fields to mappers
7. Auto deselection of secure fields
8. Schedule

## View Job Status

1. Display job stats by job and project
2. Ability to view the failure message
3. Run the failed job

## Upload Sensitive Data

1. Upload CSV with sensitive data elements
2. Add sensitive data elements
3. Select project optionally

## Upload Sensitive Data

1. Preview data loaded ( a sample)

Database  
Design

User	Roles	Project	DataSources	Job	JobRuns	DataElementExclusions
UserId	Id	ProjectId	DataSourceId	JobId	JobRunId	Id
Name	RoleName	Name	Type	ProjectId	JobId	Name
Password			Config (JsonB)	Name	JobConfig(JsonB)	Value
				Type	LogPath	Type (sensitive, exclusion)
UserRoles				Status	Status	
	UserId			JobConfig (JsonB)	Result (JsonB)	
	RoleId			CreatedBy	CreatedTime	
	ProjectId			CreatedTime	UpdatedTime	
				UpdatedTime		

Core Framework: Django framework  
Database: Postgres

Conceptually this is an ETL tool. We will use ETL concepts to create and obfuscate the data while moving from one data source to another.

Key components of the data generation include

1. Mappers - Passthrough, link, mask, randomize, synthetic - These are small units to support the transformation part of the load process. These should be built independently to support transformation at the row level. Multiple mappers can be chained to perform the transformations.
2. Primary Key hash creation for data mapping. A hashing algorithm to create a hash from a single or multiple values.
3. Scheduler - To schedule the jobs created and run accordingly
4. Data readers and writers - These are to support the extract and load parts of the ETL process