

# System Design Document

Friday October 8, 2021

*Sprint 1*



*Krutik Tejalkumar Shah*

*Mark (LiangShu) Chen*

*Steven Hans Linantoro*

*Tanzim Ahmed*

*Tony Chen*

*Pasa Aslan*

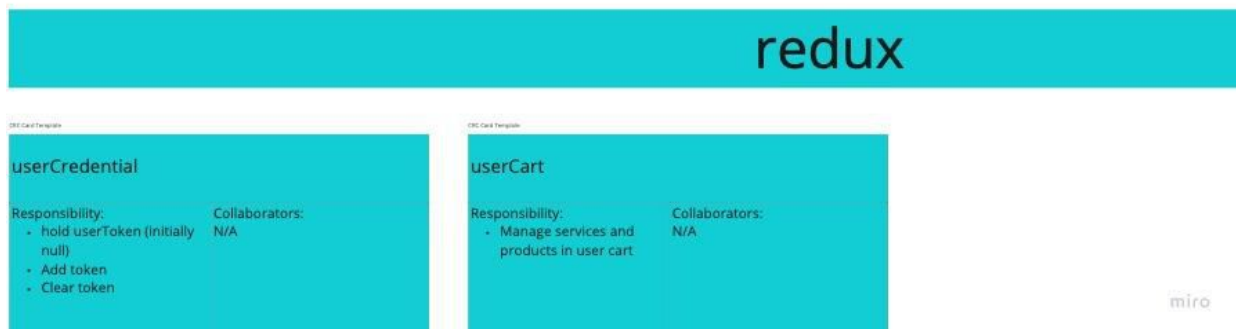
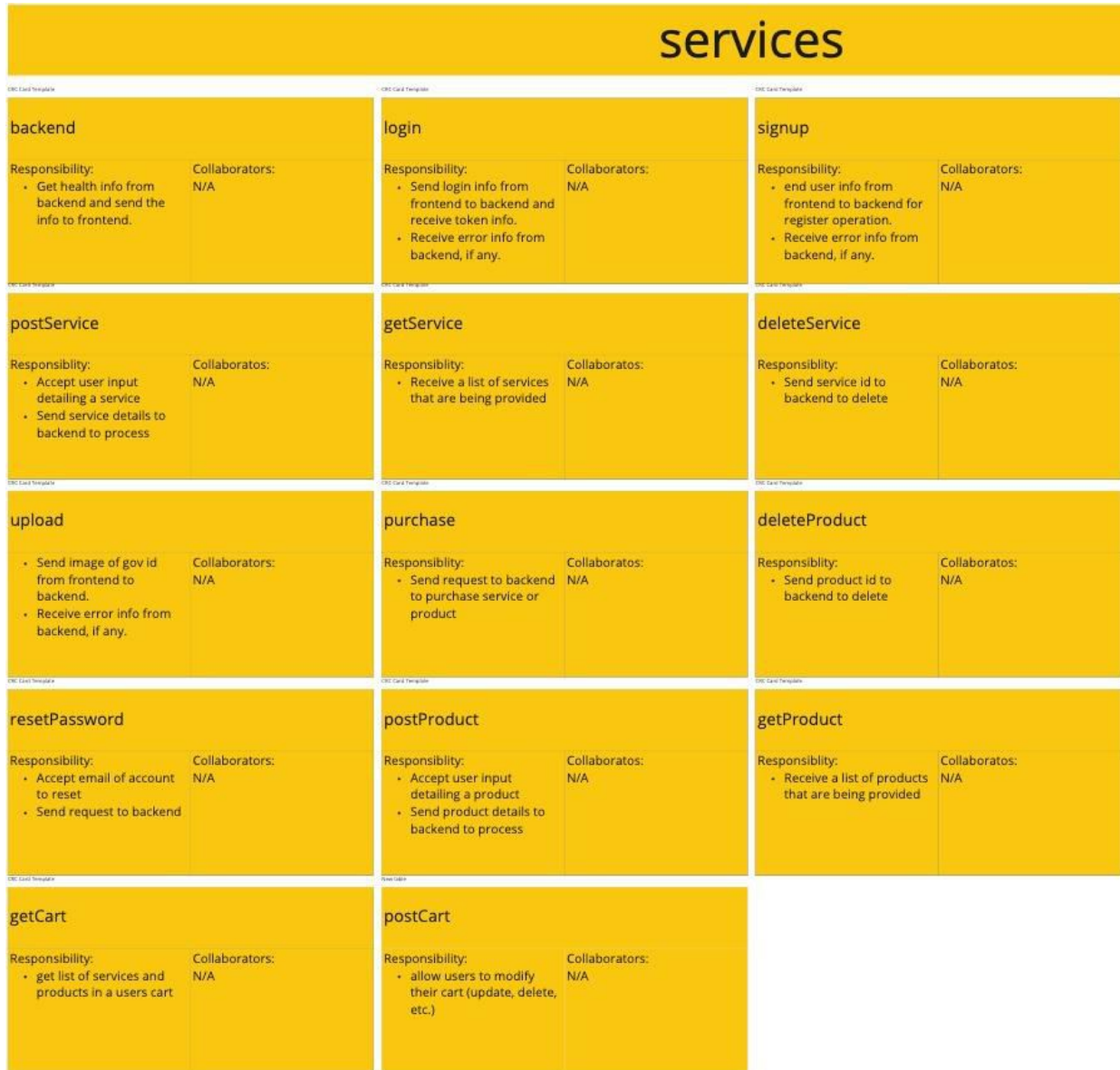
*Payam Yektamaram*

# Table of Contents

<b>CRC Cards</b>	<b>2</b>
Frontend	2
Backend	4
<b>System Architecture Diagram (Node.js Layered Arch)</b>	<b>6</b>
Frontend	6
Architecture	6
Backend	7
Server Endpoints & Descriptions	7
Database Schema Models	8
Architecture	9

# CRC Cards

## Frontend



# screens

OSC Card Template

## SignInScreen

### Responsibility:

- Allow user to input email and password
- Validate user credentials
- Authenticate user session
- Link to Sign-Up.
- Link to Forgot-Password.

### Collaborators:

- login
- userCredential

OSC Card Template

## SignUpScreen

### Responsibility:

- Accept email, username, and password to register new user
- Validate user input
- Link to SignInScreen

### Collaborators:

- signup

OSC Card Template

## ResetPasswordScreen

### Responsibility:

- Accept email of account to reset password
- Accept new password

### Collaborators:

- resetPassword

OSC Card Template

## HomeTabScreen

### Responsibility:

- Provide detailed search functionality for services

### Collaborators:

- N/A

OSC Card Template

## ServicesTabScreen

### Responsibility:

- List available services
- Provide filtering functionality
- Allow verified users to post their own service
- Allow user to view a specific service in more detail
- Allow user to delete their own service
- Allow user to purchase service

### Collaborator:

- postService
- getService
- deleteService
- purchase

OSC Card Template

## StoreTabScreen

### Responsibility:

- List available product
- Provide filtering functionality
- Allow user to view a specific product in more detail
- Allow admin to delete product
- Allow user to purchase product

### Collaborators:

- postProduct
- getProduct
- deleteProduct
- purchase

OSC Card Template

## AccountTabScreen

### Responsibility:

- List user info
- Allow user to upload government id
- Allow user to log out

### Collaborators:

- upload

OSC Card Template

## CartTabScreen

### Responsibility:

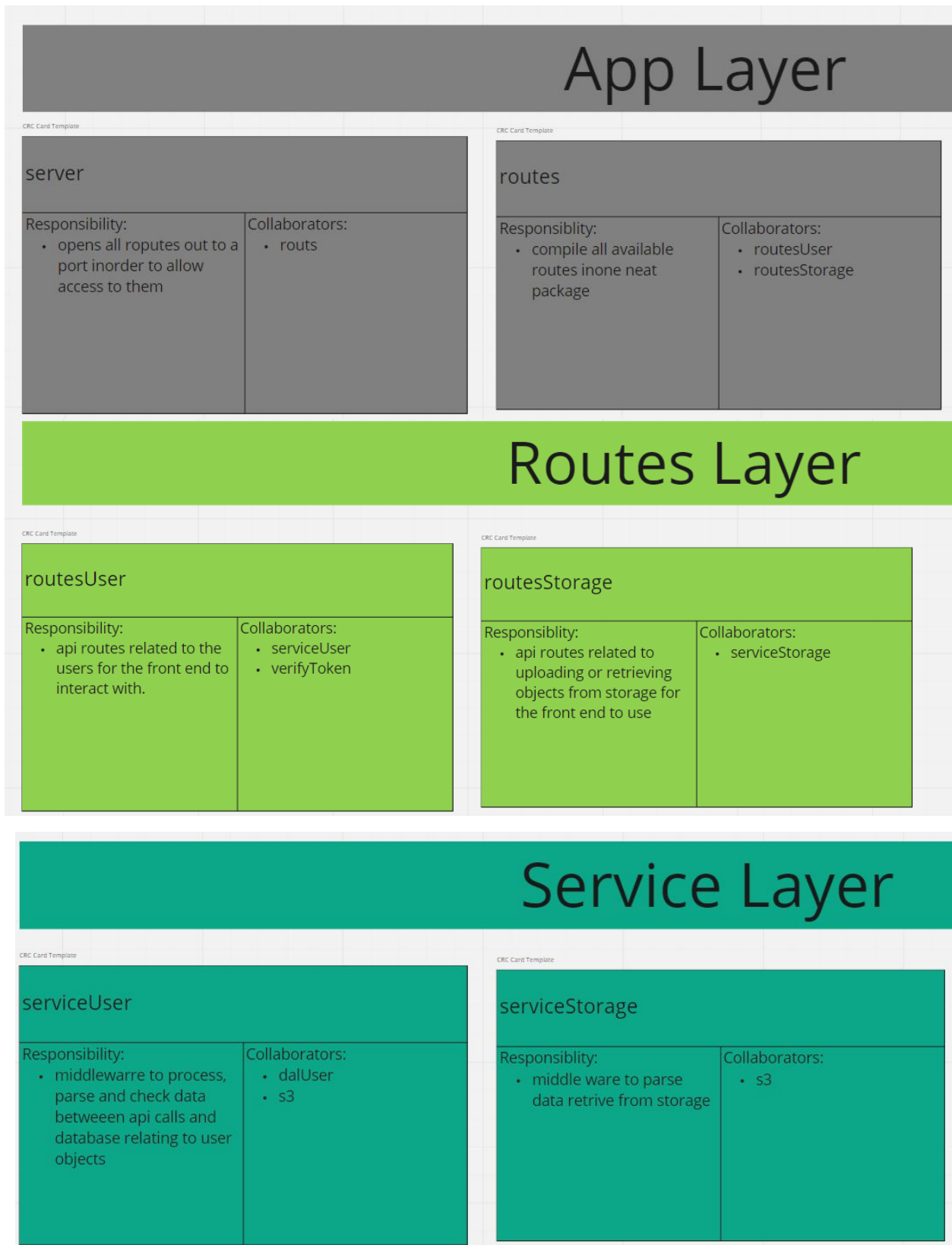
- List services and products in the users cart
- Allow user to manage their cart

### Collaborators:

- getCart
- postCart
- userCart

miro

## Backend



# Data Access Layer

CDC Card Template

## dalUser

### Responsibility:

- upload and query user related objects from the database
- utilfunction related to user objects

### Collaborators:

modelUser  
modelUserCredentials  
modelVerificationRequests

# Models

CDC Card Template

## modelUser

### Responsibility:

- define User object schema

### Collaborators:

N/A

CDC Card Template

## modelUserCredentials

### Responsibility:

- define UserCredential object schema

### Collaborators:

N/A

CDC Card Template

## modelVerificationRequest

### Responsibility:

- Define VerificationRequest object schema

### Collaborators:

N/A

# Utils

CDC Card Template

## s3

### Responsibility:

- upload and retrieve objects from s3 bucket

### Collaborators:

N/A

CDC Card Template

## verifyToken

### Responsibility:

- verify session token

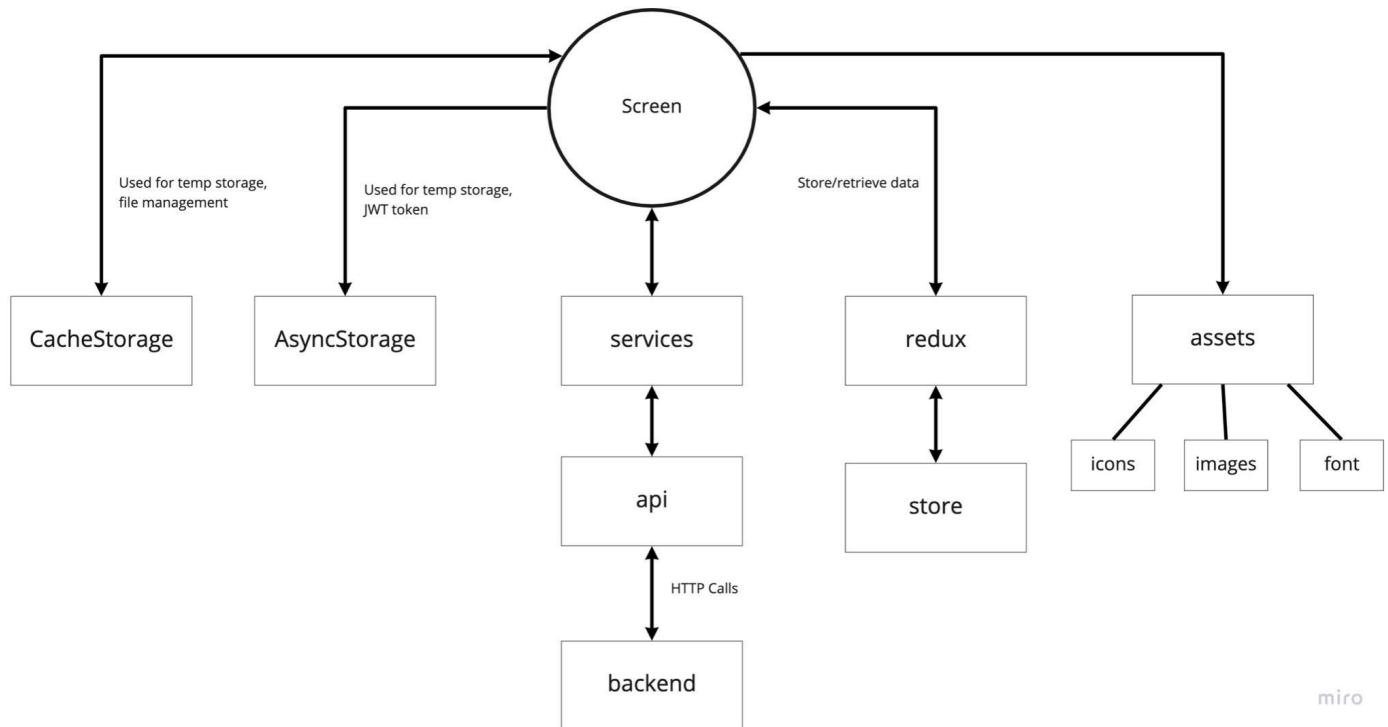
### Collaborators:

N/A

# System Architecture Diagram ([Node.js Layered Arch](#))

## Frontend

### Architecture



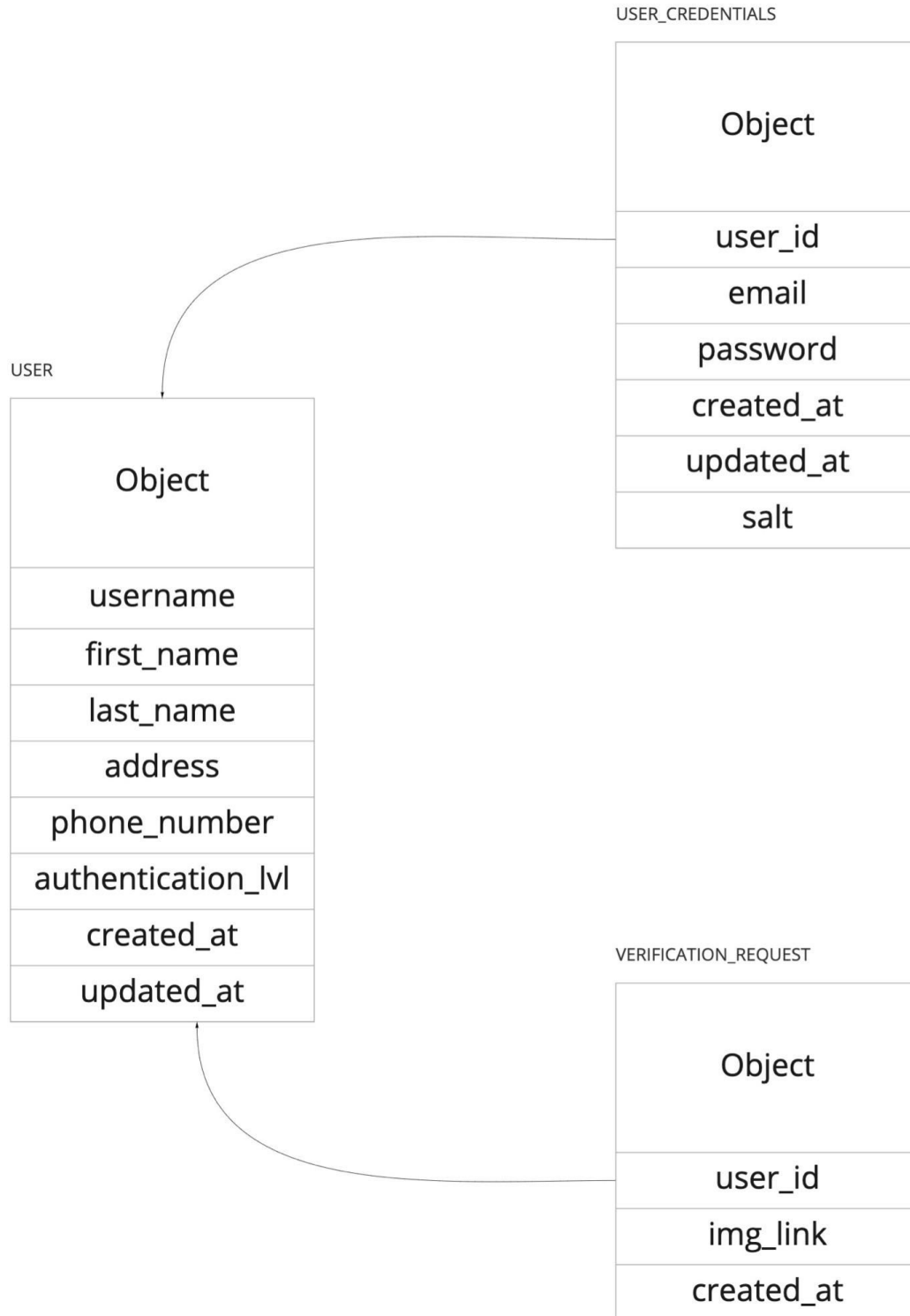
# Backend

## Server Endpoints & Descriptions

Summary	Path	Auth Required	Verification level Required	Method	Payload	Response
Register	/api/v1/users/register	No	None	POST	{ email: email, username: user, password: password }	USER
Login	/api/v1/users/login	No	None	POST	{ email: email, password: password }	USER
Forgot Password	/api/v1/forgot-password/{email}	No	None	POST	email: email	None
Reset Password	/api/v1/reset-password/{email}	No	None	POST	{ encryptedEmail: email, encryptedOTPid: id, otp: code, password: password }	USER
Update Account Details	/api/v1/users/self	Yes	User	PUT	{ first_name: f_name, last_name: l_name, address: address, phone: phone }	USER
Request Verification	/api/v1/users/self/request-verification	Yes	Unverified User	POST	gov-id: png/jpg	None
Retrieve Verification	/api/v1/verification-request	Yes	Owner/Admin	GET		List[Verification_request]
Approve	/api/v1/verification-request	Yes	Owner/Admin	PUT	{ user_id: user_id, approved: boolean }	None
Get Account Details	/api/v1/users/self	Yes	User	GET	None	USER
Get Account Details By Id	/api/v1/users/{user_id}	No	None	GET	None	USER
Get Image	/api/v1/storage/media/{image_key}	No	None	GET	None	png/jpg



## Database Schema Models



## Architecture

