

personal_access_tokens Laravel/Sanctum package default		
PK	id	int, UNSIGNED, AI
	tokenable_type	varchar(255), NOT NULL
FK	tokenable_id	NOT NULL, ON UPDATE RESTRICT, ON DELETE CASCADE
	name	varchar(255), NOT NULL
	token	varchar(64), NOT NULL
	abilities	text, NULLABLE
	last_used_at	timestamp, NULLABLE
	expires_at	timestamp, NULLABLE

users		
PK	id	int, UNSIGNED, AI
	first_name	varchar(30), NOT NULL
	last_name	varchar(30), NOT NULL
	email	varchar(255), NOT NULL, UNIQUE
	email_verified_at	timestamp, NULLABLE
	password	varchar(255), NOT NULL
	phone	varchar(30), NOT NULL

loans		
PK	id	int, UNSIGNED, AI
FK	user_id	NOT NULL, ON UPDATE RESTRICT, ON DELETE CASCADE
	amount	int, UNSIGNED, NOT NULL
	remained_principle	decimal(15,2), UNSIGNED, NOT NULL
	term	small int, UNSIGNED, NOT NULL
d	payment_period	ENUM('weekly', 'monthly'), NOT NULL
	start_date	DATE, NOT NULL
	state	ENUM('approved', 'pending', 'paid'), NOT NULL, DEFAULT 'pending'

user_has_roles Unique constrain : [user_id, role_id]		
PK	id	int, UNSIGNED, AI
FK	user_id	NOT NULL, ON UPDATE RESTRICT, ON DELETE CASCADE
FK	role_id	NOT NULL, ON UPDATE RESTRICT, ON DELETE CASCADE

roles		
PK	id	int, UNSIGNED, AI
	name	varchar(30), NOT NULL

scheduled_repayments		
PK	id	int, UNSIGNED, AI
FK	loan_id	NOT NULL, ON UPDATE RESTRICT, ON DELETE CASCADE
	amount	decimal(15,2), UNSIGNED, NOT NULL
	due_date	DATE, NOT NULL
	state	ENUM('active', 'pending', 'paid'), NOT NULL, DEFAULT 'pending'

One user can have many roles like: admin, manager...

remained\_principle is the amount of principle of the loan remain that needs to be paid. When a scheduled\_payment is paid this remained\_principle has to be reduced by the amount is paid.

Use this field to keep track of the max amount of money the borrower could pay in advance. And when we should change the loans 'state' to be 'paid'