

# NFC Livestock Management System

## description:

The NFC Livestock Management System (NLMS) leverages providing precise detailed health monitoring for each animal through unique IDs. A website facilitates efficient data management, interaction, and transaction processes among breeders and buyers. This innovative system promises to significantly improve efficiency and transparency in livestock management practices and activities such as sale, purchase, and transfer of ownership between breeders and buyers.

## Model 1: Livestock

```
@NotNull(message = "id Cannot be null.")
```

```
@Size(min = 3,message = "ID must be more than 3")
```

```
private String micro_id;
```

```
@NotNull(message = "id Cannot be null.")
```

```
@Size(min = 3,message = "ID must be more than 3")
```

```
private String user_id;
```

```
@NotEmpty(message = "Can not be empty")
```

```
private String livestock_status;
```

```
@Positive(message = "must be positive number")
```

```
@NotNull(message = "price can not be null")
```

```
private float price;
```

```
@NotNull(message = "date fo brith Cannot be null. ")
```

```
@PastOrPresent(message = "should be a date in the past or the present. ")
```

```
@JsonFormat(pattern="yyyy-MM-dd")
```

```
private Date date_of_birth;
```

## Model 2: Buyer

```
@NotNull(message = "id Cannot be null.")
```

```
@Size(min = 3,message = "ID must be more than 3")
```

```
private String buyer_id;
```

```
@Pattern(regexp = "^05\\d{8}$", message = "Phone number must start with  
'05' and consist of exactly 10 digits")
```

```
private String phoneNumber;
```

```
@Email
```

```
@NotEmpty(message = "email must be not empty")
```

```
private String email;
```

```
@NotEmpty(message = "username should be not Empty")
```

```
@Size(min=4,max=10,message = "userName must size between 3-11")
```

```
private String userName;
```

```
@NotEmpty(message = "password should be not empty")
```

```
@Pattern(regexp= "^(?=.*[A-Za-z])(?=.*\\d)(?=.*[@$!%*#?&])[A-Za-  
z\\d@$!%*#?&]{8,}$",message = "error password")
```

```
private String password;
```

## Model 3: Seller

```
@NotNull(message = "id Cannot be null.")
```

```
@Size(min = 3,message = "ID must be more than 3")
```

```
private String seller_id;
```

```
@Pattern(regexp = "^05\\d{8}$", message = "Phone number must start with
```

'05' and consist of exactly 10 digits")

private String phoneNumber;

@Email

@NotEmpty(message = "email must be not empty")

private String email;

@NotEmpty(message = "password should be not empty")

@Pattern(regexp= "^(?=.\*[A-Za-z])(?=.\*\\d)(?=.\*[@\$!%\*#?&])[A-Za-z\\d@\$!%\*#?&]{8,}\$",message = "error password")

private String password;

@NotEmpty(message = "username should be not Empty")

@Size(min=4,max=10,message = "userName must size between 3-11")

private String userName;

@NotNull(message = "id Cannot be null.")

@Size(min = 3,message = "ID must be more than 3")

private String livestock\_id;

## Model 4: Micro

@NotNull(message = "id Cannot be null.")

@Size(min = 3,message = "ID must be more than 3")

private String micro\_id;

## Model 5: Admin

@Size(min = 3,message = "ID must be more than 3")

@NotEmpty(message = "admin id can not to be Empty ")

```
private String admin_id;
```

```
@NotEmpty(message = "username should be not Empty")
```

```
@Size(min=4,max=10)
```

```
private String userName;
```

```
@NotEmpty(message = "password should be not empty")
```

```
@Pattern(regexp= "^(?=.*[A-Za-z])(?=.*\\d)(?=.*[@$!%*#?&])[A-Za-z\\d@$!%*#?&]{8,}$",message = "error password")
```

```
private String password;
```

## Model 6: Transaction

```
@Size(min = 3,message = "ID must be more than 3")
```

```
@NotEmpty(message = " id can not to be Empty ")
```

```
private String transaction_id;
```

```
@Size(min = 3,message = "ID must be more than 3")
```

```
@NotEmpty(message = " id can not to be Empty ")
```

```
private String livestock_id;
```

```
@Positive(message = "must be positive number")
```

```
@NotNull(message = "price can not be null")
```

```
private float price;
```

```
@Size(min = 3,message = "ID must be more than 3")
```

```
@NotEmpty(message = " id can not to be Empty ")
```

```
private String id_seller;
```

```
@Size(min = 3,message = "ID must be more than 3")
```

```
@NotEmpty(message = " id can not to be Empty ")
```

```
private String id_buyer;
```

```
@NotNull(message = "transaction date Cannot be null. ")
@PastOrPresent(message = "should be a date in the past or the present. ")
@JsonFormat(pattern="yyyy-MM-dd")
private Date transcation_date;

@NotNull(message = "Time cannot be null")
@Pattern(regexp = "^([01]?[0-9])2[0-3]:[0-5][0-9]$", message = "Time must
be in the format HH:mm")
private String transcation_time;
```

Name:Shahad Alhazmi