

IBM Project Report On

Intelligent bank loan application and Isolate Sensitive Data in your databases

Developed By: -

Kevin
Pandya(18082271004)
Rahul
Prajapati(18162171024)
Om Patel(18162121013)

Guided By:-

Prof. Rahul Shrimali(Internal)
Mr. Nagendra Sir (External)

**Submitted to
Department of Computer Science & Engineering
Institute of Computer Technology**



Year: 2022



CERTIFICATE

This is to certify that the **IBM** Project work entitled “**Intelligent bank loan application and Isolate Sensitive Data in your databases**” by Kevin Pandya(Enrolment No.18082271004), Om Patel (Enrolment No.18162121013) and Rahul Prajapati (EnrolmentNo.18162171024) of Ganpat University, towards the partial fulfillment of requirements of the degree of Bachelor of Technology – Computer Science and Engineering, carried out by them in the CSE(CBA/BDA/CS) Department at Elegant Microweb Pvt. Ltd.. The results/findings contained in this Project have not been submitted in part or full to any other University / Institute for award of any other Degree/Diploma.

Name & Signature of Internal Guide

Name & Signature of Head

Place: ICT - GUNI

Date:

ACKNOWLEDGEMENT

IBM Internship project is a golden opportunity for learning and self-development. I consider myself very lucky and honored to have so many wonderful people lead me through in completion of this project. First and foremost, I would like to thank Dr. Hemal Shah, Principal, ICT, and Prof. Dharmesh Darji , Head, ICT who gave us an opportunity to undertake this project. My grateful thanks to Prof. abc & Mr.xyz (Internal & External Guides) for their guidance in project work Predicting Application Rating of Google Play Store, who despite being extraordinarily busy with academics, took time out to hear, guide and keep us on the correct path. We do not know where we would have been without his/her help. CSE department monitored our progress and arranged all facilities to make life easier. We choose this moment to acknowledge their contribution gratefully.

ABSTRACT

The ONLINE LOAN MANAGEMENT SYSTEM is an exceptionally productive interaction to deal with all credit related exchanges in an extremely precise and advantageous manner. Interface which offices a client to apply for an advance on the web and to follow the status every now and then. Utilizing it's a paperless framework so responsibility is diminished .It is extremely useful for the administrator.

INDEX

- 1. Introduction**
- 2. Project Scope**
- 3. Software & Hardware Requirements**
- 4. Project Model (Flowchart)**
- 5. Project Plan**
- 6. Implementation Details**
- 7. Code for Training our Model from the Collected Datasets**
- 8. Current Output**
- 9. Conclusion & Future Work**
- 10. References**

CHAPTER: 1 INTRODUCTION

CHAPTER 1 INTRODUCTION

In the Online Loan Management System project, clients can apply for advances on the web. Clients need to enter credit applications on the web. Staff or administrators has position to support or reject advanced applications. Clients can see their Loan account subtleties, Interest rate, reimbursement plan subtleties, and so forth. Clients can make credit installments on the web. After the installment the framework creates a receipt with all out paid sum and equilibrium sum. The framework ascertains handling charge and late installment expense consequently. Chairmen can see installment subtleties, credit account subtleties, forthcoming installment subtleties, ended records, and so on.

CHAPTER: 2 PROJECT SCOPE

CHAPTER 2 PROJECT SCOPE

- This framework is intended to handily keep up with the information of the advanced clients explicitly. Clients can apply for a credit without visiting the bank.
- Clients can apply for a credit account on the web. Clients need to fill their prerequisites in the advance application.
- This framework is made to keep the records about the clients who have taken an advance from a bank.
- This framework permits clients to make installments on the web.
- The chairman is the fundamental client of this web application and he can add worker subtleties, Loan types, punishment charges, and so forth
- Various types of Reports can be seen.

CHAPTER: 3 SOFTWARE AND HARDWARE REQUIREMENTS

CHAPTER 3 SOFTWARE AND HARDWARE REQUIREMENTS

Minimum Hardware Requirements

| | |
|------------------|---------|
| Processor | 2.0 GHz |
| RAM | 4GB |
| HDD | 40GB |

Table 3.1 Minimum Hardware Requirements

Minimum Software Requirements

| | |
|-------------------------------|--|
| Operating System | Any operating system which can support an internet browser. |
| Programming language | Python Django Framework for creating UI MySQL server as Database AWS/IBM Cloud |
| Other tools & tech | Internet browser (further- Amazon Web Services like S3, EC2, EBS) |

Table 3.2 Minimum Software Requirements

CHAPTER: 4 PROCESS MODEL

CHAPTER 4 PROCESS MODEL

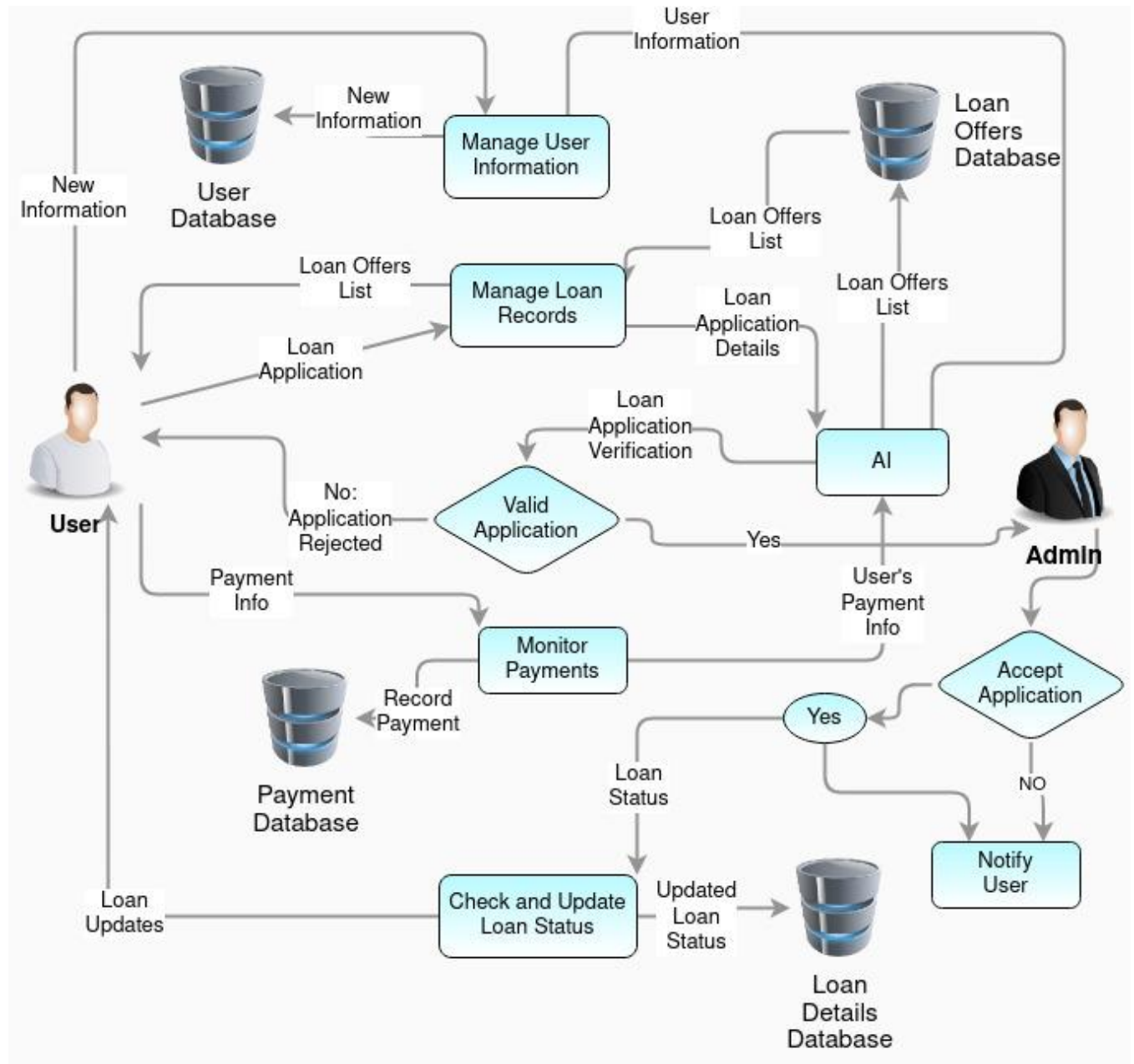


Figure 4.1 Process Model of Project

CHAPTER: 5 PROJECT PLAN

CHAPTER 5 PROJECT PLAN

5.1 Approach

Foster an easy to understand GUI

Foster Separate secure Login Portals for credit banks and advance appliers

Set up secure and secluded information exchanges of both advanced banks and credit candidates information.

Set up Credit Score Calculating System for Loan Applications

Normal Loan Application Interface with Available Interest Rates and offers

Application Acceptance and Rejection Interface from Loan Lenders

Trading Loan Agreement Report Interface

CHAPTER: 6 IMPLEMENTATION DETAILS

CHAPTER 6 IMPLEMENTATION DETAIL

Language - Python 3
Frame work - Django Web Framework
IDE :- Visual Studio
Operating System – Linux
Database:- Sqlite3
Domain :- localhost

Code:-

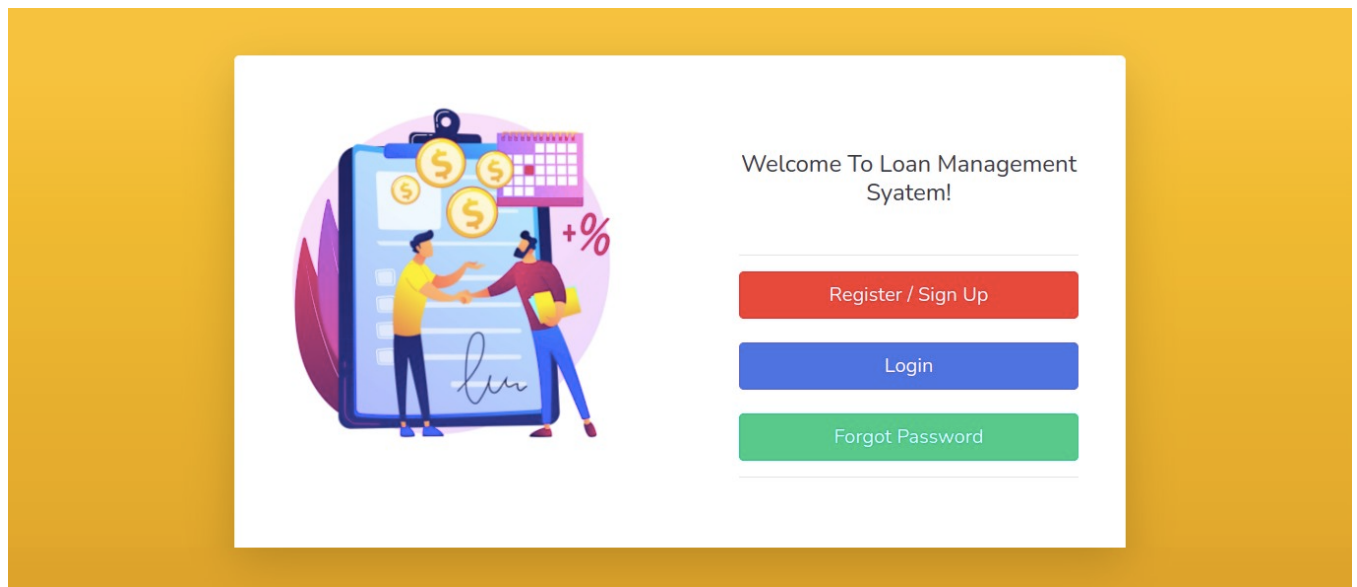


Figure 6.1 home page (Done by OM Patel)

Signup (Done By OM Patel) :-

```
class SignUpForm(UserCreationForm):  
  
    first_name = forms.CharField(max_length=30, required=False,  
help_text='Optional')  
    last_name = forms.CharField(max_length=30, required=False,  
help_text='Optional')  
    email = forms.EmailField(max_length=254, help_text='Enter a valid email  
address')  
  
    class Meta:  
        model = User  
        fields = [  
            'username',
```

```
'first_name',  
'last_name',  
'email',  
'password1',  
'password2',  
]
```

Output:-

Create an Account!

Username*

Required. 150 characters or fewer. Letters, digits and @/./+/-_ only.

First name

Optional

Last name

Optional

Email*

Enter a valid email address

Password*

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

Password confirmation*

Enter the same password as before, for verification.

[Submit](#)

[Forgot Password?](#)

[Already have an account? Login!](#)

[Home](#)

Figure 6.2 signup page

Login() (Done by Kevin Pandya) :-

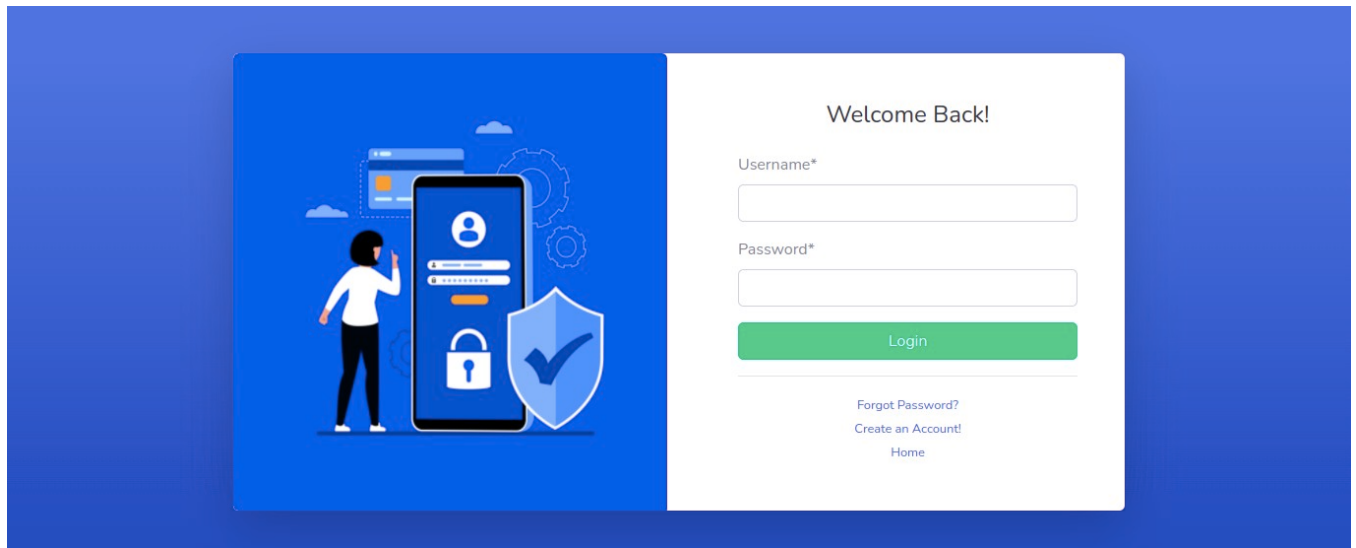


Figure 6.3 login page

Forgot Password (Done by Rahul Prajapati):-

```
# Forget Password

path('password-reset/',
     auth_views.PasswordResetView.as_view(
         template_name='common/password-reset/password_reset.html',
         subject_template_name='common/password-reset/password_reset_subject.txt',
         email_template_name='common/password-reset/password_reset_email.html',
         # success_url='/login/'
     ),
     name='password_reset'),
path('password-reset/done/',
     auth_views.PasswordResetDoneView.as_view(
         template_name='common/password-reset/password_reset_done.html'
     ),
     name='password_reset_done'),
path('password-reset-confirm/<uidb64>/<token>',
     auth_views.PasswordResetConfirmView.as_view(
         template_name='common/password-reset/password_reset_confirm.html'
```

```

    ),
    name='password_reset_confirm'),
path('password-reset-complete/',
    auth_views.PasswordResetCompleteView.as_view(
        template_name='common/password-reset/password_reset_complete.html'
    ),
    name='password_reset_complete'),

path('oauth/', include('social_django.urls', namespace='social')), # <--
here
]

```

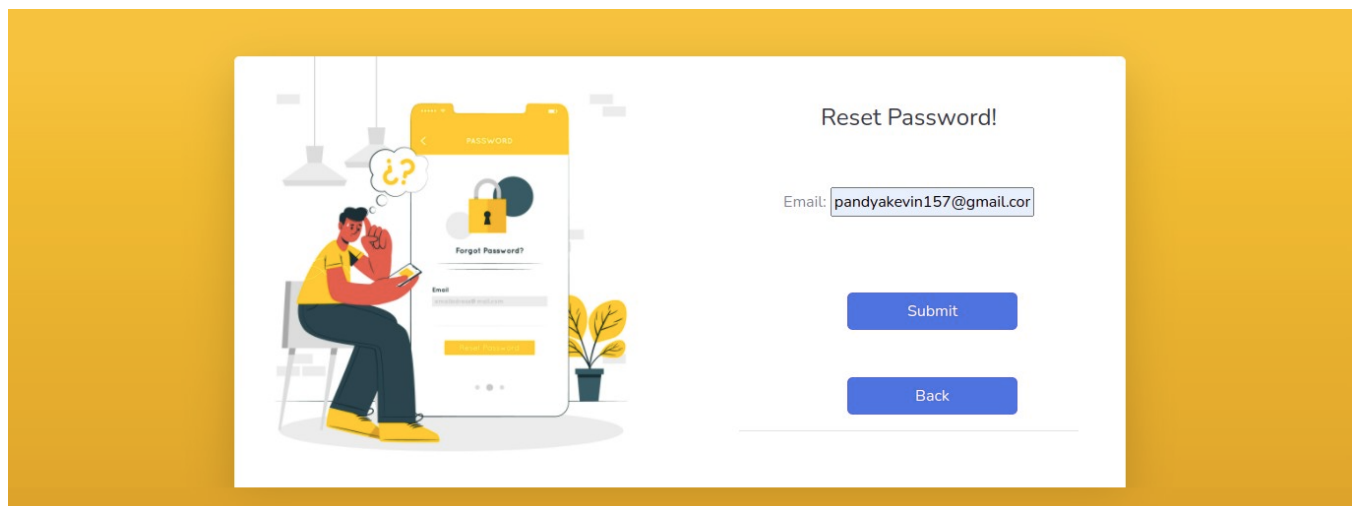


Figure 6.4: Reset Password Page

Reset Password

We have sended you reset password link on your email. Please Check Your Email.

[Back to Login Page](#)

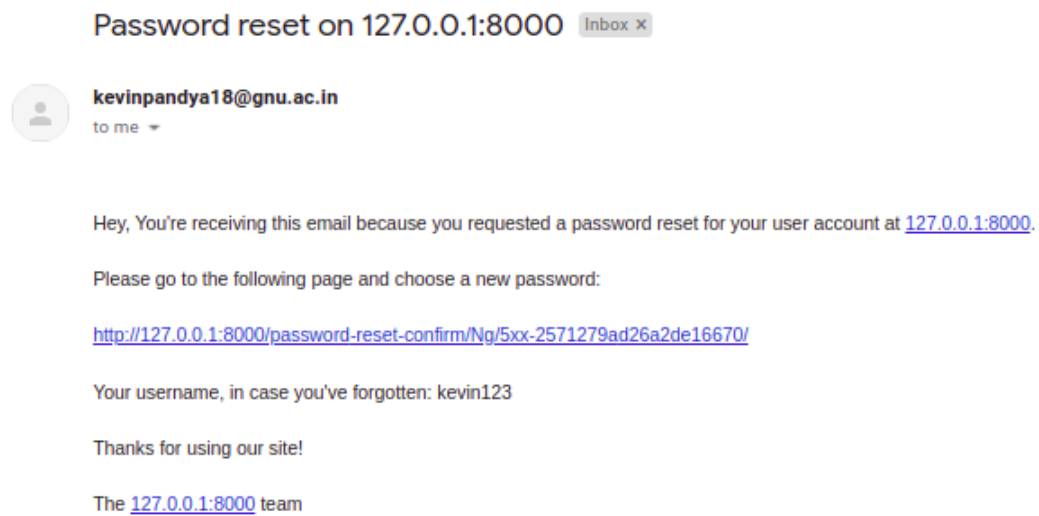


Figure 6.5: Password Reset Email

Master Page (Done by OM Patel, Kevin Pandya, Rahul Prajapati):-

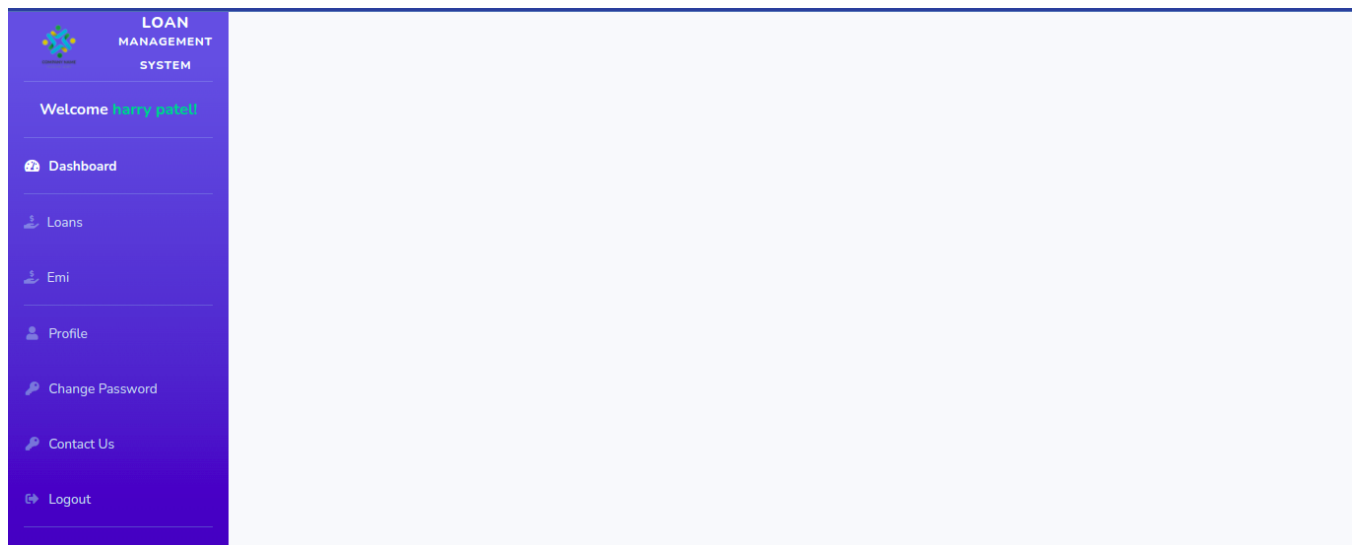


Figure 6.6: Master Page

Loan Page (Done by Kevin Pandya):-

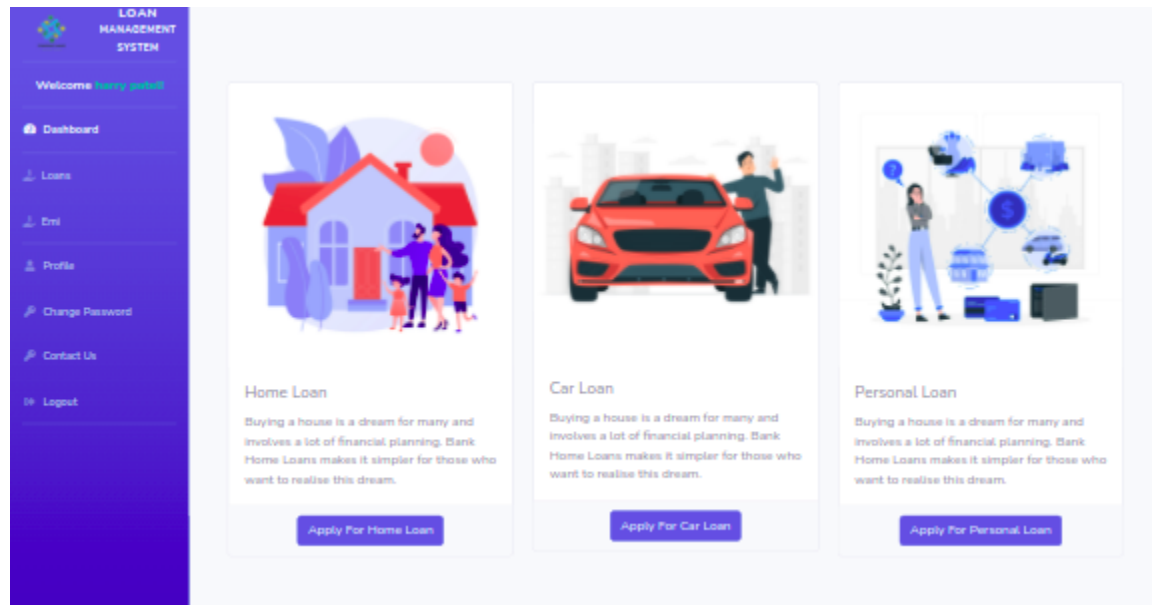



Figure 6.7: Loans Page

EMI Calculator (Done by OM Patel) :-

```
function emi () {if (document.getElementById('loan1').value==null ||
document.getElementById('loan1').value.length==0 ||
document.getElementById('months1').value==null ||
document.getElementById('months1').value.length==0 ||
document.getElementById('rate1').value==null ||
document.getElementById('rate1').value.length==0 )
{document.getElementById('pay1').value='Data Reqd.'}else {var pay1='';var
princ1= document.getElementById('loan1').value;var term1=
document.getElementById('months1').value;var
intr1=document.getElementById('rate1').value /
1200;document.getElementById('pay1').value =Math.round(princ1 * intr1 /
(1-(Math.pow(1/(1 + intr1), term1)))*100)/100;
document.getElementById('gt1').value=
Math.round((document.getElementById('pay1').value *
document.getElementById('months1').value)*100)/100;document.getElementById('tin
tt1').value=Math.round((document.getElementById('gt1').value*1 -
document.getElementById('loan1').value*1)*100)/100}}
```



**LOAN
MANAGEMENT
SYSTEM**

Welcome **Harry jaisil**

- Dashboard
- Loans
- Emi
- Profile
- Change Password
- Logout

EMI Calculator

| | |
|--|--|
| Loan Amount | <input type="text" value="200000"/> |
| Repayment in months | <input type="text" value="36"/> |
| Interest Rate | <input type="text" value="8"/> |
| <input type="button" value="Reset"/> <input type="button" value="Submit"/> | |
| EMI | <input type="text" value="6267.27"/> |
| Interest payable | <input type="text" value="25621.72"/> |
| Total payable | <input type="text" value="225621.72"/> |

Figure 6.8: EMI Calculator

Profile Page (Done by Rahul Prajapati):-

```

class ProfileUpdateView(LoginRequiredMixin, TemplateView):

    user_form = UserForm

    profile_form = ProfileForm

    template_name = 'common/profile-update.html'

    def post(self, request):

        post_data = request.POST or None
        file_data = request.FILES or None

        user_form = UserForm(post_data, instance=request.user)
        profile_form = ProfileForm(post_data, file_data,
instance=request.user.profile)

        if user_form.is_valid() and profile_form.is_valid():
            user_form.save()
            profile_form.save()
            messages.error(request, 'Your profile is updated successfully!')
            return HttpResponseRedirect(reverse_lazy('profile'))

```



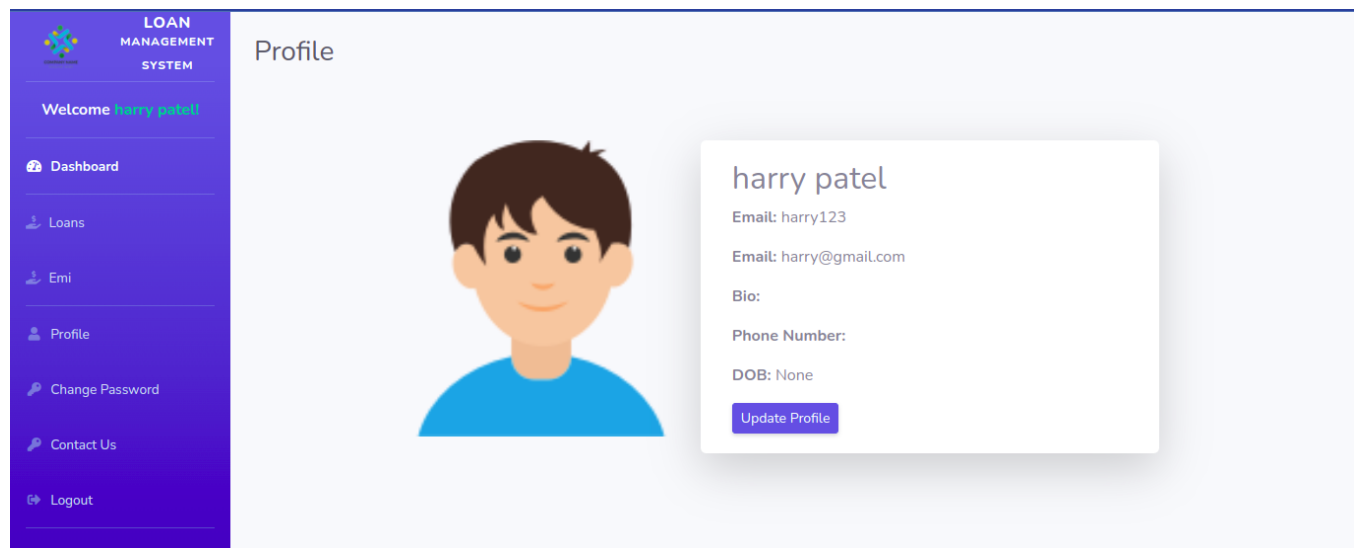
```

context = self.get_context_data(
    user_form=user_form,
    profile_form=profile_form
)


return self.render_to_response(context)

def get(self, request, *args, **kwargs):
    return self.post(request, *args, **kwargs)

```



Profile Update (Done by Rahul Prajapati):-

**LOAN
MANAGEMENT
SYSTEM**

Welcome **harry patel**

Dashboard

Loans

Emi

Profile

Change Password

Contact Us

Logout

Profile Update

Update Your Profile!

Username*

harry123

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

First name

harry

Last name

Kevin

Email address

harry@gmail.com

Bio

Updated bio

Phone number

(092) 838-8474

Birth date

2000-10-27

Profile image

Currently: [default-avatar.png](#) ☐ Clear


Change:

Choose file

compass (1).png

Update

[Back](#)



LOAN
MANAGEMENT
SYSTEM

Welcome harry patel

Dashboard

Loans

Emi

Profile

Change Password

Contact Us

Logout

Profile



harry patel

Email: harry123

Email: harry@gmail.com


Bio: Updated bio

Phone Number: 923848481111

DOB: Oct. 28, 2000

Update Profile

Change Pwd (Done by Rahul Prajapati):-



LOAN
MANAGEMENT
SYSTEM

Welcome harry patel

Dashboard

Loans

Emi

Profile

Change Password

Contact Us

Logout

Change Password

Change Your Password!

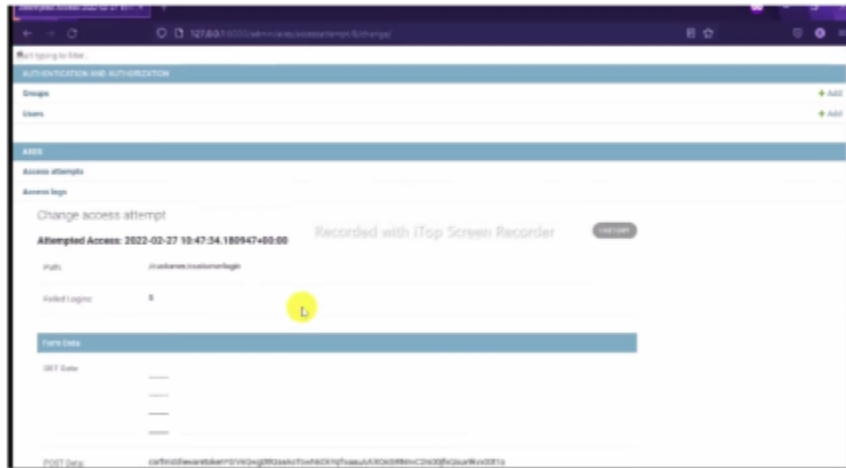
Old password*

New password*

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

New password confirmation*

Loan Failure:-



```
AXES_ENABLED = True

AXES_FAILURE_LIMIT = 5

AXES_COOLOFF_TIME = timedelta(minutes=3)

AXES_ONLY_USER_FAILURES = True

# AXES_LOCKOUT_TEMPLATE = 'template name'
```

Contact-us (Done by Kevin Pandya):-

LOAN MANAGEMENT SYSTEM

Welcome [Admin](#)

Dashboard

Loans

EMI

Profile

Change Password

Contact Us

Logout

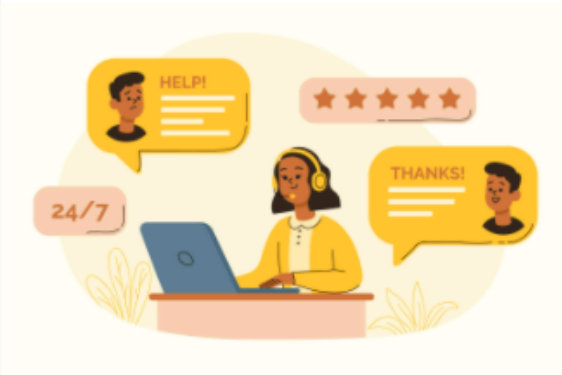
Language

Select Language

English Hindi Tamil

Contact Us

Wed Mar 09 2022 14:52:56 GMT+0530 (India Standard Time)



Contact Form!

Please write your problem below if you have any questions. One of our staff members will be happy to contact you directly and answer your questions as soon as possible.

Problem

Describe your problem here

Send To Admin

```
def admin_question_view(request):  
    questions = models.Question.objects.all()  
    return render(request, 'loan/admin_question.html', {'questions': questions})
```

```
def ask_question_view(request):  
    customer = models.Customer.objects.get(user_id=request.user.id)  
    questionForm=CFORM.QuestionForm()  
  
    if request.method=='POST':  
        questionForm=CFORM.QuestionForm(request.POST)  
        if questionForm.is_valid():  
            question = questionForm.save(commit=False)  
            question.customer=customer  
            question.save()
```

```

        return redirect('question-history')

    return
render(request, 'customer/ask_question.html', {'questionForm': questionForm, 'customer': customer})

//admin
def contactus_view(request):
    sub = forms.ContactusForm()

    if request.method == 'POST':
        sub = forms.ContactusForm(request.POST)

        if sub.is_valid():
            email = sub.cleaned_data['Email']
            name=sub.cleaned_data['Name']
            message = sub.cleaned_data['Message']

            send_mail(str(name)+' || '+str(email),message,settings.EMAIL_HOST_USER, settings.EMAIL_RECEIVING_USER,
fail_silently = False)

            return render(request, 'loan/contactussuccess.html')

        return render(request, 'loan/contactus.html', {'form':sub})

```

LOAN MANAGEMENT SYSTEM

Welcome **Jack Patel!**

- Dashboard
- Loans
- Emi
- Profile
- Change Password
- Contact Us
- Logout
- Language

Select Language

Asked Question History To Admin

Wed Mar 09 2022 15:07:53 GMT+0530 (India Standard Time)

| Sr No. | Question | Admin Comment | Asked Date |
|--------|--------------|---------------|---------------|
| 1 | I have doubt | Hello | March 9, 2022 |
| 2 | Doubt | Nothing | March 9, 2022 |

```
def update_question_view(request,pk):
    question = models.Question.objects.get(id=pk)
    questionForm=forms.QuestionForm(instance=question)

    if request.method=='POST':
        questionForm=forms.QuestionForm(request.POST,instance=question)

        if questionForm.is_valid():

            admin_comment = request.POST.get('admin_comment')
            question = questionForm.save(commit=False)
            question.admin_comment=admin_comment
            question.save()

            return redirect('admin-question')

    return
render(request,'loan/update_question.html',{'questionForm':questionForm})
```

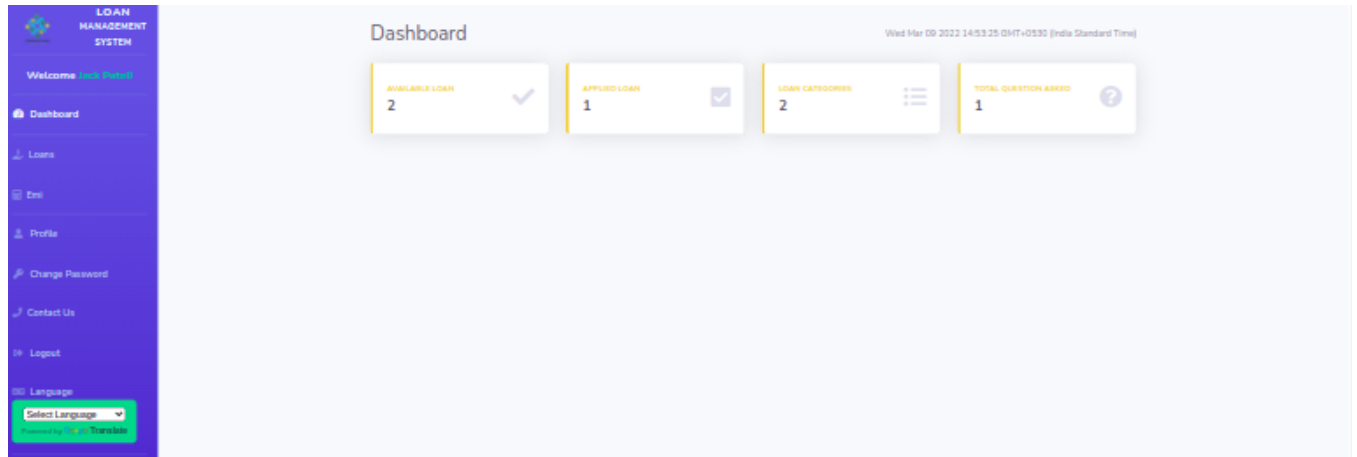
```
def question_history_view(request):
    customer = models.Customer.objects.get(user_id=request.user.id)
    questions = CMODEL.Question.objects.all().filter(customer=customer)
```

```

        return
    render(request, 'customer/question_history.html', {'questions': questions, 'customer': customer})

```

Dashboard of client()



```

@login_required(login_url='customerlogin')
def customer_dashboard_view(request):
    dict={
        'customer':models.Customer.objects.get(user_id=request.user.id) ,
        'available_loan':CMODEL.Loan.objects.all().count() ,

        'applied_loan':CMODEL.LoanRecord.objects.all().filter(customer=models.Customer.
objects.get(user_id=request.user.id)).count() ,

        'total_category':CMODEL.Category.objects.all().count() ,

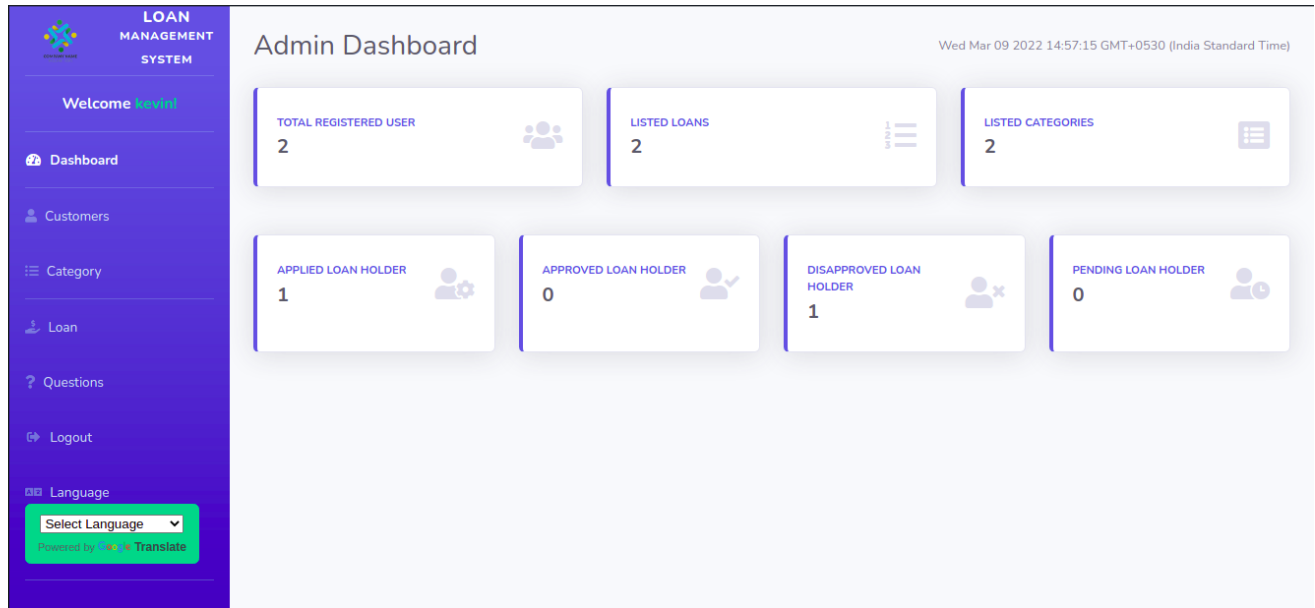
        'total_question':CMODEL.Question.objects.all().filter(customer=models.Customer.
objects.get(user_id=request.user.id)).count() ,

    }

    return render(request, 'customer/customer_dashboard.html', context=dict)

```

Dashboard Of Admin()



```
@login_required(login_url='adminlogin')
def admin_dashboard_view(request):
    dict={
        'total_user':CMODEL.Customer.objects.all().count(),
        'total_loan':models.Loan.objects.all().count(),
        'total_category':models.Category.objects.all().count(),
        'total_question':models.Question.objects.all().count(),
        'total_loan_holder':models.LoanRecord.objects.all().count(),

        'approved_loan_holder':models.LoanRecord.objects.all().filter(status='Approved')
    ).count(),

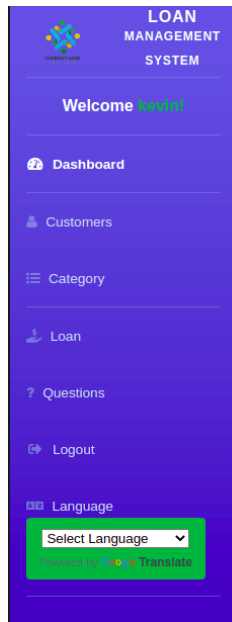
        'disapproved_loan_holder':models.LoanRecord.objects.all().filter(status='Disapproved')
    ).count(),

        'waiting_loan_holder':models.LoanRecord.objects.all().filter(status='Pending')
    ).count(),

    }

    return render(request, 'loan/admin_dashboard.html', context=dict)
```

Customer list()



Customers

Wed Mar 09 2022 14:57:46 GMT+0530 (India Standard Time)

| Sr No. | Name | Profile Picture | Contact | Address | Update | Delete |
|--------|-----------------|---|------------|-----------|---|---|
| 1 | Jack Patel |  | 1111111111 | ahmedabad |  |  |
| 2 | Rahul Prajapati |  | 1234567890 | kjnj |  |  |

```
@login_required(login_url='adminlogin')
def admin_view_customer_view(request):
    customers= CMODEL.Customer.objects.all()
    return
render(request,'loan/admin_view_customer.html',{'customers':customers})

@login_required(login_url='adminlogin')
def update_customer_view(request,pk):
    customer=CMODEL.Customer.objects.get(id=pk)
    user=CMODEL.User.objects.get(id=customer.user_id)
    userForm=CFORM.CustomerUserForm(instance=user)
    customerForm=CFORM.CustomerForm(request.FILES,instance=customer)
    mydict={'userForm':userForm,'customerForm':customerForm}
    if request.method=='POST':
        userForm=CFORM.CustomerUserForm(request.POST,instance=user)
        customerForm=CFORM.CustomerForm(request.POST,request.FILES,instance=customer)
        if userForm.is_valid() and customerForm.is_valid():
```

```
        user=userForm.save()

        user.set_password(user.password)

        user.save()

        customerForm.save()

        return redirect('admin-view-customer')

    return render(request, 'loan/update_customer.html', context=mydict)


@login_required(login_url='adminlogin')
def delete_customer_view(request, pk):

    customer=CMODEL.Customer.objects.get(id=pk)


    user=User.objects.get(id=customer.user_id)

    user.delete()

    customer.delete()

    return HttpResponseRedirect('/admin-view-customer')
```

Add Loan Category

LOAN
MANAGEMENT
SYSTEM

Welcome **kevin!**

[Dashboard](#)

[Customers](#)

[Category](#)


[Loan](#)

[Questions](#)

[Logout](#)

[Language](#)

Select Language

Powered by  Translate

Admin Dashboard

Wed Mar 09 2022 14:58:09 GMT+0530 (India Standard Time)


VIEW CATEGORY

ADD CATEGORY

UPDATE CATEGORY

DELETE CATEGORY

View Category

LOAN
MANAGEMENT
SYSTEM

Welcome **kevin!**

[Dashboard](#)

[Customers](#)

[Category](#)


[Loan](#)

[Questions](#)

[Logout](#)

[Language](#)

Select Language

Powered by  Translate


Categories

Wed Mar 09 2022 14:58:55 GMT+0530 (India Standard Time)

| Sr No. | Category Name | Creation Date |
|--------|---------------|---------------|
| 1 | Home Loan | March 8, 2022 |
| 2 | Car Loan | March 9, 2022 |

```
def admin_view_category_view(request):  
    categories = models.Category.objects.all()  
    return  
render(request, 'loan/admin_view_category.html', {'categories':categories})
```

Add Category



**LOAN
MANAGEMENT
SYSTEM**


Welcome **Kevin!**

- Dashboard
- Customers
- Category
- Loan
- Questions
- Logout
- Language

Select Language

Loan Category

Wed Mar 09 2022 14:59:16 GMT+0530 (India Standard Time)



Add Category!

Please Insert Category Of Loan Want To add.

Category Name

Add

```
def admin_add_category_view(request):
    categoryForm=forms.CategoryForm()


    if request.method=='POST':
        categoryForm=forms.CategoryForm(request.POST)

        if categoryForm.is_valid():
            categoryForm.save()

            return redirect('admin-view-category')

    return
render(request, 'loan/admin_add_category.html', {'categoryForm':categoryForm})
```

Update Category


**LOAN
MANAGEMENT
SYSTEM**

Welcome **kevin!**

Dashboard

Customers

Category

Loan

Questions

Logout

Language

Select Language

[English](#)
[Hindi](#)
[Gujarati](#)
[Urdu](#)

Update Category

Wed Mar 09 2022 14:59:42 GMT+0530 (India Standard Time)

| Sr No. | Category Name | Creation Date | Update Category |
|--------|---------------|---------------|-------------------|
| 1 | Home Loan | March 8, 2022 | ✍ |
| 2 | Car Loan | March 9, 2022 | ✍ |

UPDATE CATEGORY

Category Name

Home Loan

Update

```
@login_required(login_url='adminlogin')
def update_category_view(request,pk):
    category = models.Category.objects.get(id=pk)
    categoryForm=forms.CategoryForm(instance=category)

    if request.method=='POST':
        categoryForm=forms.CategoryForm(request.POST,instance=category)

        if categoryForm.is_valid():
```

```


        categoryForm.save()

        return redirect('admin-update-category')

    return
render(request, 'loan/update_category.html', {'categoryForm': categoryForm})

```

Delete Category



**LOAN
MANAGEMENT
SYSTEM**

Welcome **kevin!**



- Dashboard
- Customers
- Category
- Loan
- Questions
- Logout
- Language

Select Language

Powered by Google Translate

Delete Category

Wed Mar 09 2022 15:00:35 GMT+0530 (India Standard Time)

| Sr No. | Category Name | Creation Date | Delete Category |
|--------|---------------|---------------|---|
| 1 | Home Loan | March 8, 2022 |  |
| 2 | Car Loan | March 9, 2022 |  |


```

def admin_delete_category_view(request):
    categories = models.Category.objects.all()

    return
render(request, 'loan/admin_delete_category.html', {'categories': categories})

```

Admin Questions




**LOAN
MANAGEMENT
SYSTEM**

Welcome **kevin!**



- Dashboard
- Customers
- Category
- Loan
- Questions
- Logout
- Language

Select Language

Powered by  Google Translate

Questions

Wed Mar 09 2022 15:01:16 GMT+0530 (India Standard Time)


| Sr No. | Customer | Problem | Reply | Asked Date | Update |
|--------|----------|--|---------|---------------|---|
| 1 | Jack | I have doubt | Hello | March 9, 2022 |  |
| 2 | Rahul | I have a simple question about this... | Nothing | March 9, 2022 |  |

```
def admin_question_view(request):
    questions = models.Question.objects.all()
    return render(request, 'loan/admin_question.html', {'questions': questions})
```

Welcome **kevin!**

- Dashboard
- Customers
- Category
- Loan
- Questions
- Logout
- Language

Select Language

Powered by  Google Translate

Question!

Please give the response to the client entering input below

Question By Client

I have doubt

Revert Back

Enter Your Reply To Client

Update

```
def update_question_view(request, pk):
    question = models.Question.objects.get(id=pk)
    questionForm=forms.QuestionForm(instance=question)
```



```

if request.method=='POST':

    questionForm=forms.QuestionForm(request.POST,instance=question)

    if questionForm.is_valid():

        admin_comment = request.POST.get('admin_comment')

        question = questionForm.save(commit=False)

        question.admin_comment=admin_comment

        question.save()

        return redirect('admin-question')

    return
render(request,'loan/update_question.html',{'questionForm':questionForm})

```

View Loan


**LOAN
MANAGEMENT
SYSTEM**

Welcome **kevin!**

- Dashboard
- Customers
- Category
- Loan
- Questions
- Logout
- Language

Select Language


Google Translate

Loans

Wed Mar 09 2022 15:02:10 GMT+0530 (India Standard Time)

| Sr No. | Loan Name | Category | Creation Date |
|--------|-----------|-----------|---------------|
| 1 | Home Loan | Home Loan | March 8, 2022 |
| 2 | Car Loan | Car Loan | March 9, 2022 |

Add Loan



**LOAN
MANAGEMENT
SYSTEM**

Welcome **Kevin!**

- Dashboard
- Customers
- Category
- Loan
- Questions
- Logout
- Language


Select Language

English

Translate

Loan Add

Wed Mar 09 2022 15:02:45 GMT+0530 (India Standard Time)



Add Loan!

Please Insert Loan selecting the category of loan need to add and insert name.

Category

Car Loan

Loan Name

Car Loan


Add

```
def admin_add_loan_view(request):
    loanForm=forms.LoanForm()

    if request.method=='POST':
        loanForm=forms.LoanForm(request.POST)
        if loanForm.is_valid():
            categoryid = request.POST.get('category')
            category = models.Category.objects.get(id=categoryid)
            loan = loanForm.save(commit=False)
            loan.category=category
            loan.save()
            return redirect('admin-view-loan')

    return render(request,'loan/admin_add_loan.html',{'loanForm':loanForm})
```

Update Loan


**LOAN
MANAGEMENT
SYSTEM**

Welcome **Kevin!**

- Dashboard
- Customers
- Category
- Loan
- Questions
- Logout
- Language

Select Language

English

Translate

Update Loan

Wed Mar 09 2022 15:03:11 GMT+0530 (India Standard Time)

| Sr No. | Loan Name | Category | Creation Date | Update |
|--------|-----------|-----------|---------------|-------------------|
| 1 | Home Loan | Home Loan | March 8, 2022 | ✎ |
| 2 | Car Loan | Car Loan | March 9, 2022 | ✎ |

```

@login_required(login_url='adminlogin')
def update_loan_view(request,pk):
    loan = models.Loan.objects.get(id=pk)
    loanForm=forms.LoanForm(instance=loan)

    if request.method=='POST':
        loanForm=forms.LoanForm(request.POST,instance=loan)

        if loanForm.is_valid():

            categoryid = request.POST.get('category')
            category = models.Category.objects.get(id=categoryid)


            loan = loanForm.save(commit=False)
            loan.category=category
            loan.save()

            return redirect('admin-update-loan')

    return render(request,'loan/update_loan.html',{'loanForm':loanForm})

```

Delete Loan


**LOAN
MANAGEMENT
SYSTEM**

Welcome **kevin!**



- Dashboard
- Customers
- Category
- Loan
- Questions
- Logout
- Language

Select Language

Translate


Delete Loan

Wed Mar 09 2022 15:04:27 GMT+0530 (India Standard Time)

| Sr No. | Loan Name | Category | Creation Date | Delete |
|--------|-----------|-----------|---------------|---|
| 1 | Home Loan | Home Loan | March 8, 2022 |  |
| 2 | Car Loan | Car Loan | March 9, 2022 |  |

```
def admin_delete_loan_view(request):  
    loans = models.Loan.objects.all()  
    return render(request, 'loan/admin_delete_loan.html', {'loans':loans})  
  
def delete_loan_view(request,pk):  
    loan = models.Loan.objects.get(id=pk)  
    loan.delete()  
    return redirect('admin-delete-loan')
```

Loan Applied


**LOAN
MANAGEMENT
SYSTEM**

Welcome **kevin!**

- Dashboard
- Customers
- Category
- Loan
- Questions
- Logout
- Language

Select Language

Translate

Loan Holder Record

Wed Mar 09 2022 15:04:49 GMT+0530 (India Standard Time)

| Sr No. | Loan | Customer | Applied Date | Status | Action |
|--------|------|----------|---------------|-------------|-------------|
| 1 | None | Jack | March 9, 2022 | Disapproved | Disapproved |

```
def admin_view_loan_holder_view(request):
```

```

        loanrecords = models.LoanRecord.objects.all()

        return
    render(request, 'loan/admin_view_loan_holder.html', {'loanrecords': loanrecords})

```

Approved Loan



**LOAN
MANAGEMENT
SYSTEM**

Welcome **kevin!**

- Dashboard
- Customers
- Category
- Loan
- Questions
- Logout
- Language

Select Language

Loan Holder Record **Approved**

Wed Mar 09 2022 15:05:14 GMT+0530 (India Standard Time)

| Sr No. | Loan | Customer | Applied Date | Status |
|--------|------|----------|--------------|--------|
|--------|------|----------|--------------|--------|

```

def admin_view_approved_loan_holder_view(request):

    loanrecords = models.LoanRecord.objects.all().filter(status='Approved')

    return
    render(request, 'loan/admin_view_approved_loan_holder.html', {'loanrecords': loanrecords})

def approve_request_view(request, pk):

    loanrecords = models.LoanRecord.objects.get(id=pk)

    loanrecords.status='Approved'

    loanrecords.save()

    return redirect('admin-view-loan-holder')

```

Disapproved Loan

LOAN MANAGEMENT SYSTEM

Welcome **kevin!**

- Dashboard
- Customers
- Category
- Loan
- Questions
- Logout
- Language
 - Select Language
 - Google Translate


Loan Holder Record **Disapproved**

Wed Mar 09 2022 15:05:32 GMT+0530 (India Standard Time)

| Sr No. | Loan | Customer | Applied Date | Status |
|--------|------|----------|---------------|--------------------|
| 1 | None | Jack | March 9, 2022 | Disapproved |

```
def admin_view_disapproved_loan_holder_view(request):  
    loanrecords = models.LoanRecord.objects.all().filter(status='Disapproved')  
    return  
render(request, 'loan/admin_view_disapproved_loan_holder.html', {'loanrecords': loanrecords})  
  
def disapprove_request_view(request, pk):  
    loanrecords = models.LoanRecord.objects.get(id=pk)  
    loanrecords.status='Disapproved'  
    loanrecords.save()  
    return redirect('admin-view-loan-holder')
```

Pending Loan


**LOAN
MANAGEMENT
SYSTEM**

Welcome **kevin!**

Dashboard

Customers

Category




Loan

Questions

Logout

Language

Select Language


Loan Holder Record Pending

Wed Mar 09 2022 15:05:46 GMT+0530 (India Standard Time)

| Sr No | Loan | Customer | Applied Date | Status |
|-------|------|----------|--------------|--------|
|-------|------|----------|--------------|--------|

```
def admin_view_waiting_loan_holder_view(request):
    loanrecords = models.LoanRecord.objects.all().filter(status='Pending')
    return
render(request, 'loan/admin_view_waiting_loan_holder.html', {'loanrecords': loanrecords})
```

Apply Loan Customer


**LOAN
MANAGEMENT
SYSTEM**

Welcome **Jack Patel!**

Dashboard

Loans

Emi

Profile




Change Password

Contact Us

Logout

Language


Select Language

Available Loans

Wed Mar 09 2022 15:06:28 GMT+0530 (India Standard Time)

| Sr No. | Loan Name | Category | Creation Date | Action |
|--------|-----------|-----------|---------------|--------|
| 1 | Home Loan | Home Loan | March 8, 2022 | Apply |
| 2 | Car Loan | Car Loan | March 9, 2022 | Apply |



LOAN
MANAGEMENT
SYSTEM

Welcome Jack Patel!

Dashboard

Loans

Eml

Profile

Change Password

Contact Us

Logout

Language

History

Wed Mar 09 2022 15:06:51 GMT+0530 (India Standard Time)

| Sr No. | Loan Name | Applied Date | Status |
|--------|-----------|---------------|-------------|
| 1 | None | March 9, 2022 | Disapproved |
| 2 | None | March 9, 2022 | Pending |

CHAPTER: 7 CONCLUSION AND FUTURE WORK

CHAPTER 7 CONCLUSION AND FUTURE WORK

Conclusion

This product gives an easy and user friendly User Interface where he will get intelligent suggestions for his loan management and loan applications.

This product makes the loan sanctioning fast and efficient by adding AI which will verify application details and documents and if it finds something wrong it will reject the application by itself.

All the transactions done will be secure.

Future work

- Loan Approval Mechanism with AI/ML

CHAPTER: 8 REFERENCES

CHAPTER 8 REFERENCES

- 1) <https://www.leadsquared.com/loan-management-system-features-benefits/>
- 2) <https://lendfoundry.com/faq/why-should-you-use-a-cloud-based-loan-management-software/>
- 3) <https://lendfoundry.com/blog/10-features-of-the-perfect-loan-management-software/>
- 4) <https://www.arttha.com/choosing-a-loan-management-system-for-your-bank-key-features-to-consider-for-an-optimized-experience/>
- 5) <https://www.leadsquared.com/loan-management-system-features-benefits/>