

Graded Project

Web Development using Flask

Domain:

- Finance and Banking.

Context:

Dream Housing Finance company deals in all home loans. They have presence across all urban, semi urban and rural areas. Customers first apply for a home loan after that company manually validates the customer eligibility for loan.

Company wants to automate the loan eligibility process based on customer detail provided while filling the details online.

They need a web application where a user can access their website and register, login, and enter the required details such as Gender, Marital Status, Education, Number of Dependents, Income, Loan Amount, Credit History and others for checking the eligibility for the home loan.

Project Objective:

1) This is a standard supervised classification task. A classification problem where we have to predict whether a customer is eligible for loan or not based on a given set of independent variable(s).

2) To build a Python Flask ML application where a user has to get registered by entering the username and password and login to the website and then enter their details to check whether they are eligible for loan or not.

Dataset Description:

Dataset can be found [here](#)

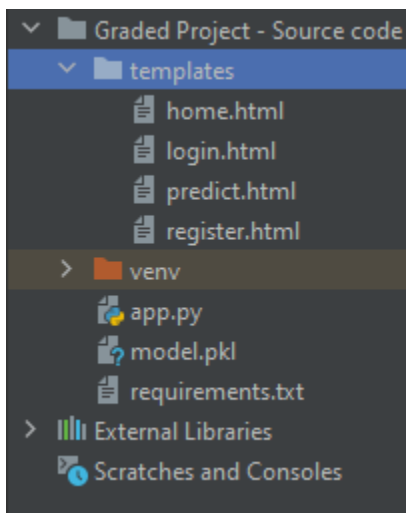
| SL. No | Attribute | Description |
|--------|----------------------|---|
| 1. | Loan ID | Unique Loan ID |
| 2. | Gender | Male or Female |
| 3. | Married | Applicant married (Y/N) |
| 4. | Dependents | Number of dependents |
| 5. | Self employed | Self employed (Y/N) |
| 6. | Education | Graduate/Undergraduate |
| 7. | Applicant Income | Applicant income (in dollars) |
| 8. | Co Applicant Income | Co Applicant Income (in dollars) |
| 9. | Loan Amount | Loan amount in thousands (in dollars) |
| 10. | Loan Amount Term | Term of loan in months |
| 11. | Credit History | Credit history meets guidelines Yes/No(1/0) |
| 12. | Property area | Urban/Semi Urban/Rural |
| 13. | Loan Status (Target) | Loan Approved (Y/N) |

Steps to the project: [Total score: 15 points]

- Model Building and saving the model using Pickle
 1. Import required libraries and read the dataset. (2)
 2. Check the first few samples, shape, info of the data and try to familiarize yourself with different features. (2)

3. Check for missing values in the dataset, if present? handle them with appropriate methods and drop redundant features. (2)
 4. Visualize the distribution of the target column 'loan_status' with respect to various categorical features and write your observations. (2)
 5. Encode the categorical data. (2)
 6. Separate the target and independent features and split the data into train and test. (2)
 7. Build any classification model to predict the loan status of the customer and save your model using pickle. (3)
- Python Flask Application development (35 points)
 1. Creating a project and virtual environment using pycharm or visual studio code and installing the required packages (5 points)

The project structure will look like this:



2. `app.py` (30 points)

- This is our main application file that is designed to call some of the implemented APIs and methods, such as connecting to the MYSQL database and creating a table for the database, and APIs such as user to register, login, enter_details, predict, and logout.
 - a) Connect to the MYSQL database and create the class for the table.
 - `User(id, username, password)` (can create the table in the database using mysql workbench also)
 - b) **Register:** This is a register API. This should take username and password and store the details in the user database.
 - c) **Login:** This API should take the username and password of registered users and successfully log them in.
 - d) **Enter Details:** This API should render the 'predict.html' page, where the user has to enter their details to check the loan eligibility.
 - e) **Predict:** This API should render the prediction results on the HTML template.
 - f) **Logout:** This API should logout the user.

Once you run the application, it should route to the `home.html` page where the user has to register.

Please login/Signup to check loan eligibility.

Login Now

Register Now

The Register.html Page will look like this, the user enters the username and password.

Register Page

Already have an account? Log In

The username and password will get added to the database.

| Result Grid | | | |
|--------------|------|----------|--------------------------|
| Filter Rows: | | | |
| | id | username | password |
| ▶ | 1 | someone | \$2b\$12\$IEzfA3qyBAHrZ. |
| * | NULL | NULL | NULL |

Login.html page will look like this, only a registered user with a correct password can be logged in.

Login Page

[Don't have an account? Sign Up](#)

Once the user has logged in, the predict.html page will be displayed where the user needs to enter the details to get the loan eligibility status.

House Loan Eligibility Prediction

Please enter the details in order to check your house loan eligibility

Gender

married

dependents

education

self_employed

Applicant Income (\$)

Coapplicant Income (\$)

Loan amount in thousands (\$)

Loan Amount Term (in months)

Credit History

Property Area

[Press here to logout](#)

Then the application will predict the loan eligibility based on the given features, and if the user clicks on logout, it will redirect to the login page.

House Loan Eligibility Prediction

Please enter the details in order to check your house loan eligibility

Gender

married

dependents

education

self_employed

Applicant Income (\$)

Coapplicant Income (\$)

Loan amount in thousands (\$)

Loan Amount Term (in months)

Credit History

Property Area

Congrats!! you are eligible for the loan

Submissions:

- Please submit the '*Model building and saving the model using pickle*' part solution in .html or .ipynb format.
- Python Flask web application project, Kindly zip the project folder and submit your solution.
- Prepare a document and add the screenshots of the home page, register page, data stored in MYSQL, login page, and the prediction page with the predicted results.
- Upload all the required files in the GitHub repository and share the github link.