

Loan Management System for Agriculture

P SAIKUMAR GOUD¹, NEHA JHUNJHUNWALA², BEJJAM VASUNDHARA DEVI³

^{1, 2, 3} Department of Computer Science and Engineering, Sreenidhi Institute of Science and Technology, Hyderabad

Abstract— *Technology is growing rapidly, people in this world also moving to a technical world where everything they want is to be done through online. To enable the agriculture sector more efficient and to get a good production, small loan can make a big difference in agriculture sector. For the adoption of modern farm practices and techniques and technologies, the shortage of agriculture loan is one of the major problems. With the help of this project Angular Agriculture Loan, loan can be provided to the peasant farmers through internet and by uploading the valid documents. This leads farmers satisfaction that they can produce products from agriculture sector and they will maintain a good growth and have will have a possibility to reduce the risks built by the land lenders. The main aim is to provide loans to the peasant farmers with zero interest. There are many news that showcase that more people are struggling in getting loan. Even though it is a small amount they are seeking for, but need more documents lots of filling forms with much time consumption and also the process involved in offline requires enough effort but with the help of this application the loan applying time will be reduced a lot and through online mode loan can be applied to peasant farmers for zero rate of interest and for this project angular is used as front-end and spring boot is used for back-end. While applying for loan the documents should original and amount should be reasonable from the side of farmer/user.*

Indexed Terms— *Front-End, Back-End, Authorization, Model, Controller, Repository, Service.*

I. INTRODUCTION

Agriculture plays key role for every country and even small changes makes big impact in this agriculture sector, this sector helps in the growth of GDP of India. As of now this is a real time example. so, company given this project that provide loans to small and peasant farmers with zero interest rates. This is an agriculture loan application helps in applying the loan and easy helps for peasant farmers who are eligible for seeking the loan they should be truthful in providing truthful documents and reducing pressure for public in case of loan. In user-side there are three pages apply

loan, loan status and profile and in the side of admin there are two pages applied loans and loan details. Here in this site admin will check the documents uploaded by user were truthful or not and the loan person is eligible or not if all good the loan will be approved else rejected and loan should be applied through user side portal. Technologies used in this project are angular for front end(client-side) and spring boot for the back end(server-side) and MYSQL for the database. The front-end and back-end are connected using rest API's through spring boot

II. EXISTING SYSTEM

The banks and financiers are providing loan to the peasant farmers but from them they are expecting some interest percentage rates and even though farmers get loan that will be provided through offline and in rural areas banks and financiers will see farmers in a low way and when they apply for loan it will take more time to get loan amount but in this project farmers get loan amount for zero interest and loan can be applied with in less time through online itself and also they need to provide documents like aadhar, VoterID or PAN cards as address proofs which will help in the process of repayment of loans.

- Disadvantages of existing system

The admin lacks complete privilege in using the application which is the admin is also considered as different user with different privilege in using the application than the normal user. High maintain costs to the application provider from the company.

III. PROPOSED SYSTEM

This proposed system has many features to compare existing systems in this proposed system, This agriculture loan application for applying for loan for a company exclusively. In this project login and signup pages are going to use the same pages for both the

users and admin it will take less space to compare the existing systems. Here the authentication plays the main role based on the role whether he/she is a user or admin based on that it will navigate to that pages so that others can't access any other pages without logging in which will be giving by far greater security than the others present where we can access the pages without logging in. On the admin side, There are 2 divisions in this project applied loans and loan status. The admin can do crud operations where he can add, update, view, and edit the data. The user can apply for loan with the data provided by the admin side the user can upload documents and apply for loan. So, by that, people clearly understand that the proposed system has more features to compare existing system.

IV. SYSTEM DESIGN

A. System Architecture

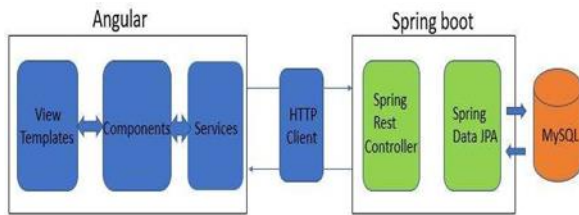


Fig 1: Architecture Diagram

Figure 1 represents to architecture diagram the architecture generally consists of 3 parts the front-end, the back-end and the database. The front-end which is where the user interacts is being done by Angular. This contains 3 types of files the view templates, the components and the service modules. These parts getting a data from the user and storing them in temporary variables this angular part is connected with the back-end i.e., spring-boot application through HTTP-client. This will help in getting data from the angular and assigning them to the back- end variables and vice-versa. The back-end consists of two parts Spring REST controller and spring data JPA. The spring REST controller helps in assigning the assigned data by the HTTP-client to the database variables. and the spring data JPA helps in query management with the database. The database helps in storing the data.

B. Data Flow Diagram

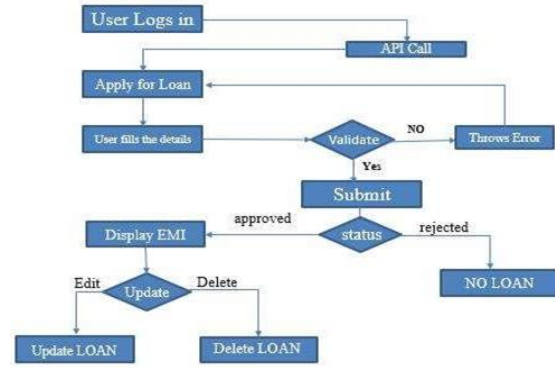


Fig 2: Data Flow Diagram

Figure 2 tells about dataflow diagram the user logs in with his credentials and the API calls and redirects him to his specific event i.e., apply loan in-case of the user and validate in-case of the admin login. And they have to fill the data for apply loan for the user and entering details for the booking in- case of the user. If all data in the fields are entered then apply for loan else throws error and directs to the select apply loan page. Then they have to reapply the loan and if admin approve loan monthly EMI is generated and visible in side of user in profile page. From that he can perform all the CRUD (Create, Read, Update, Delete) operations there.

C. Use Case Diagram

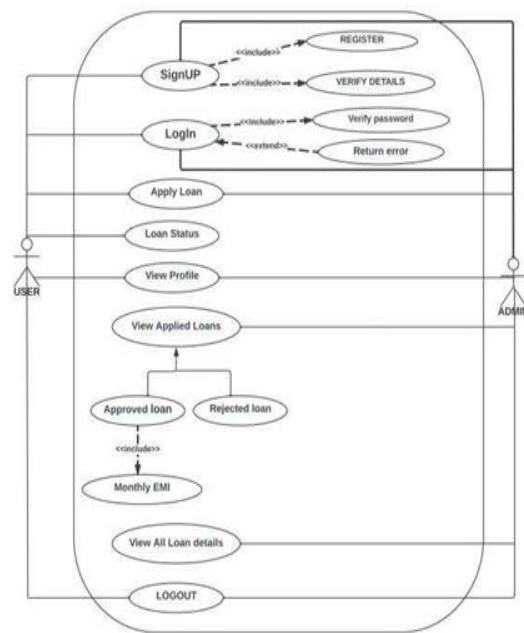


Fig 3: Use Case Diagram

Figure 3 elaborates the usecase diagram in this system there are two actors the customer and the admin. The user logs in and the authorization part verify for the valid login. If it fails then it shows invalid credentials error. After successful login the user gets to the choose apply loan page. The customer can apply loan by providing valid documents and the admin can approve or reject the loan. After approving the loan which was applied by user monthly EMI was generated and viewed in user profile in they will log out of the system. After the log out they can't again access the any of the other page again without logging in again.

D. Sequence Diagram

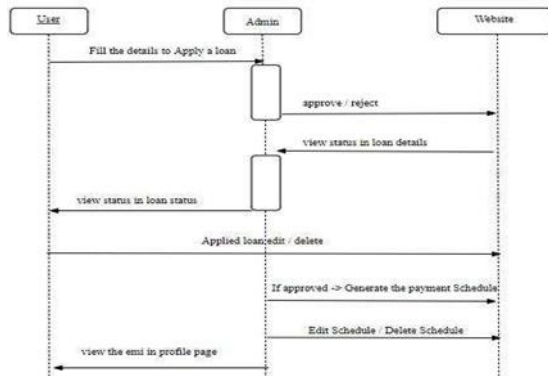


Fig 4: Sequence Diagram

Figure 4 tells about the sequence diagram in this there are three actors User, Admin, Website and each one has their own lifeline respectively and have interactions between them. In the first lifeline the user actor enquires for the applying the loan and in response to that he gets the status of loan has been approved or rejected. This interaction is done with second lifeline of the actor admin. The admin has privilege to approve to reject and view the applied loan details with interacting with the third lifeline of the website. The admin actor interacts and if the loan status is approved the monthly EMI has been generated. The second actor admin interacts with website and user to view monthly EMI in profile page.

V. IMPLIMENTATION AND RESUTS

A. Modules

1. Admin Side Front-End and Back-End creation
Initially this project developed the backend for admin side pages that is using spring-boot framework, by

creating model class, controller class, repository and ser-vice class for them. Then configure the Database for this backend in the applica- tion.properties file. Then start developing the front-end for admin side by using either template-driven form or reactive form in the format of given template. And finally test those pages with the testing pages for error finding and correction.

2. User Side Front-End and Back-End creation

In this module initially the project started developing the backend for user side pages that is using spring boot framework, by creating model class, controller class, repository and service class for them. Configure the Database for this backend in the application. properties file. Then start developing the front-end for user side by using either template-driven form or reactive form in the format of given template and finally test those pages with the testing pages for error finding and correction.

3. Login and Signup pages with back-end and front-end creation

In this third module initially start developing the backend for login and signup pages that is using spring-boot framework, by creating model class, controller class, repository and service class for them. Configure the Database for this backend in the application. properties file. Then start developing the front-end for login and signup by using either template-driven form or reactive form in the format of given template and finally test those pages with the testing pages for error finding and correction.

4. Implementing authorization and tokenization concepts for the application

Implement the tokenization on login pages by authgaurd file. Now integrating the modules and binding them with authgaurd for security. This helps in providing better security to the application by not allowing the user or the admin to access their pages without loggig in with their respective credentials.

B. Pseudo Code

- Step 1: Start
- Step 2: Admin/user need to sign-up by giving details in sign-up page.
- Step 3: Admin/user can login by using by their credentials.

Step 4: Admin can add, update, view and delete the loan in user's division which are CRUD operations for user division.

Step 5: Admin can add, update, view and delete the loan in admin division which are CRUD operations for admin division.

Step 6: Admin can approve and rejected by applied loans and generate monthly EMI if loan is approved by loan details.

Step 7: User can apply for loan by apply loan page in user side.

Step 8: User can view the applied loan status by loan status pages.

Step 9: STOP

C. INPUT AND OUTPUT

1. Input Design

Inputs for this web application are alphanumeric including of image URLs, from the starting the user has to enter his details for signing up and user needs to login to application using his credentials. Then in the case of user, user enters the details in apply loan and check status in loan status and documents in apply loan which consists of a image for each of them respectively and the admin has to approve or reject the loans in applied loans and if loan is approved can generate monthly EMI in loan details.

2. Output Design

Output for this web application will also be the same alphanumeric and images which are fetched by the URLs. After logging in and entering the data of the apply for loan there will be view page for each of them which can consider as an output page. And admin can also fetch the data for the applied loan and loan details from database and in loan details if the loan status is approved then monthly EMI will be generated and view in user profile page.

D. RESULT



Fig 5: Admin approved loan page

Figure 5 represents the admin approved loan page in this page if the loan status is approved the admin can generate a monthly EMI schedule and can edit and delete repayment schedules.

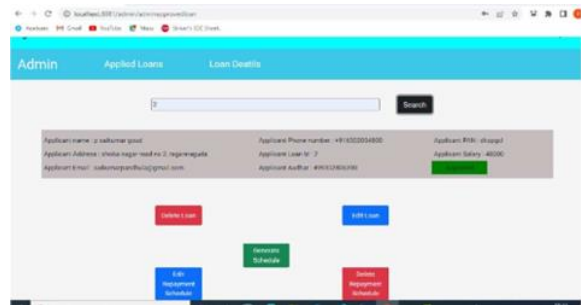


Fig 6: User loan status page

Figure 6 User loan status page tells about the user-side page which will give status about applied loan whether the loan is approved or rejected and if admin didn't verified the status will be in In-progress.

VI. CONCLUSION

This is an agriculture loan application which helps to the peasant and small farmers in applying the loan, they will apply loan by providing truthful documents like aadhar, PAN, VoterId and more in apply loan page of user-side. Admin will approve or reject the loan from applied loans page in admin-side. This is the simple full stack project for the applying loan application.

This has good security throughout the application which was helped by the authorization and tokenization, and authguard concepts of the front-end. The user can view only his related data in profile page of user-side, but the admins can see all the data

irrespective of which admin has added that data. After logging out the user or the admin can't get back to the pages without logging in again.

REFERENCES

- [1] Shylendra, H., S. 1995. Farm Loan Waivers: A Distributional and Impact Analysis of the Agricultural and Rural Debt Relief Scheme, 1990
- [2] Kanz, M. 2012. What Does Debt Relief Do for Development? Evidence from India's Bailout Program for Highly-Indebted Rural Households'
- [3] Jha, S., Bawa, K. The Economic and Environmental Outcomes of Micro finance Projects: An Indian Case Study. Environment, Development and Sustainability Vol. 9, pp. 229–239, 2007, doi: 10.1007/s10668-005- 9020-5
- [4] T. Hajji, S. Y. El Jasouli, J. Mbarki and E. M. Jaara," Microfinance risk analysis using the business intelligence," 2016 4th IEEE International Colloquium on Information Science and Technology (CiSt), Tangier, Vol.3, pp. 675-680, 2016, doi: 10.1109/CIST.2016.7804971.
- [5] Apriani Dorkas Rambu Atahau, Andrian Dolfriandra Huruta, Cheng- Wen Lee, Rural microfinance sustainability: Does local wisdom driven-governance work, Journal of Cleaner Production, Vol 267, pp.1-10, 122153, ISSN 0959-6526, 2020, doi: 10.1016/j.jclepro.2020.122153.
- [6] R. R. Mukkamala, R. Vatrapu, P. K. Ray, G. Sengupta, and S. Halder," Converging Blockchain and Social Business for Socio-Economic Development," 2018 IEEE International Conference on Big Data (BigData), Seattle, WA, USA, pp. 3039-3048, 2018, doi: 10.1109/Big- Data.2018.8622238.