Spring 2016

Illinois State Finance Corporation

Allamudi Uday Kiran
Governors State University

Pallet Sravan Kumar
Governors State University

Vecha Tarun Mithilesh
Governors State University

Kontam Vinod
Governors State University

Follow this and additional works at: http://opus.govst.edu/capstones

Part of the Databases and Information Systems Commons

Table of Contents

CPSC8985   Illinois State Finance Corporation

1 **Project Description**

1.1 *Project Abstract*

*Our project named **Illinois State Finance corporation** deals with the different types of loans* for Personal loan, car loan, crop loan, Home loan. Generally we search for companies who give loans with low interest. Our company can offer you different types of loans in single place. Users can search the details from user portal where you will aware of interest rate and how many installments. It displays the information such as cost , installments, interest, validation and estimation quote.

1.2. Competitive Information

The objective of this application is to develop a finance corporation system for Improving Software Quality and Reliability is useful for applications developed in an organization. This project can be used to reduce the calculation of interest's rates where we are able to get everything once we register for the company website.

1.3 Relationship to other applications

The existing system in does not have any information regarding interest rate in particular. Where the customers has to request for specific loan then he will be able get the details of desired loan details.

Problems in Existing System

- Process is not transparency.
- Consuming more time for users.
- Customers find difficult to get solution.

Development of New System

The development of the new system contains the following activities, which try to automate the entire process keeping in view of the database integration approach.

- This project is aimed to provide better result for customers by providing all the details in all sectors.
- We provide all the details with easy access where user can easily understand.
- Users can take decisions easily because we provide all the information and calculation easily where user will take no time to understand and decide.
- Our project is Finance Corporation where to provide loans in maximum sectors.

1.4 Assumptions and Dependencies

The proposed system has information in detail where customer has his own choice of choosing and types of payments, number of installments, These system is developed for the convenience of customers where now a days employees are paying active role in customers decisions because lack of information they are having in websites.

1.5 Future Enhancements

We can provide full access to customers, we are also planning to develop application for mobiles, iPad, tablets.

1.6 Definitions and Acronyms

SRS can be represent as Software Requirement Specifications

MMS can be represent as Meeting Scheduler System

FR can be represent as Functional Requirement

NFR can be represent as Nonfunctional requirement

JDBC can be represent as Java Database Connectivity

HTTP can be represent as Hypertext transfer Protocol

HTML can be represent as Hypertext Markup Language

2   Technical Description

As mentioned earlier in the existing system has less information in the websites. Customers has to contact customer care and able to provide all the details what the representative is required. Customers has to provide all the details to get the estimation and there is no security for the user. These are some factors which make me interested to develop these project.

2.1   Project / Application Architecture:

These project is developed with the model-2 architecture, which is a complex design pattern used by the java Web application are able to display the content from the logic can be manipulated the content. Model 2 is a architecture which drives separately for logic and display.it was assisted with the MVC architecture. The Java Blue Prints, for example, originally recommended using EJBs to encapsulate the MVC Model.
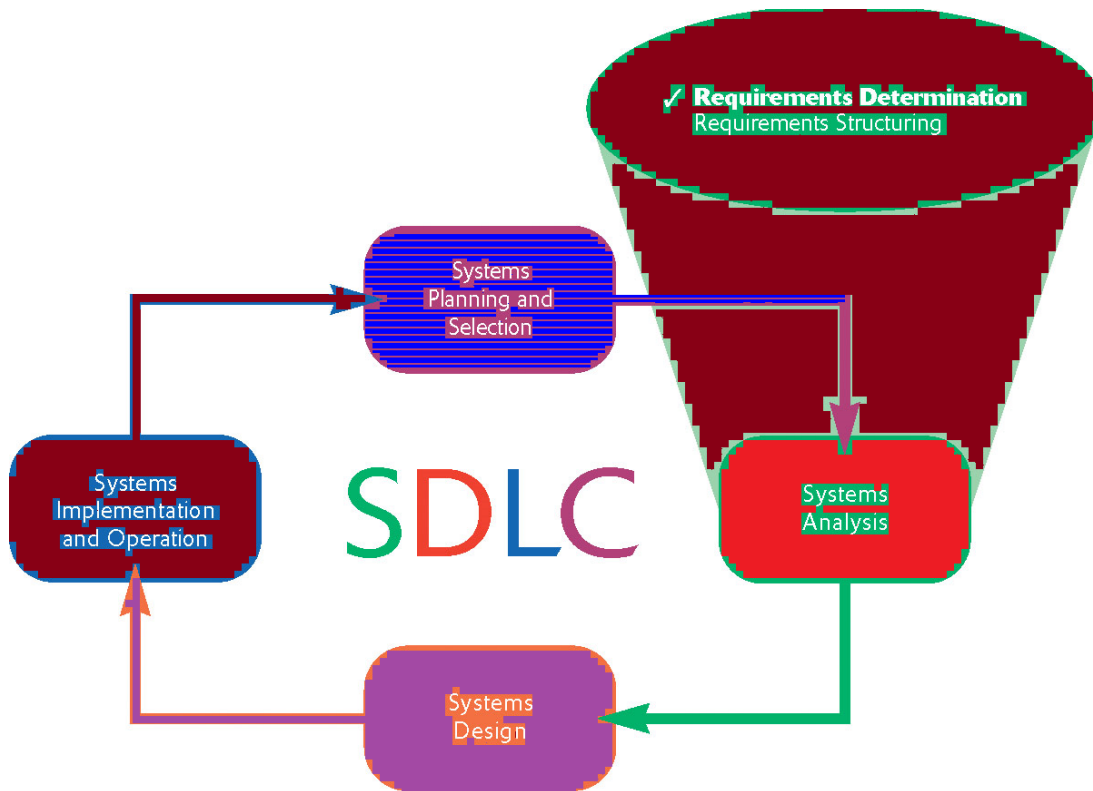
Stages in SDLC:

- Requirement
- Analysis for Project
- Designing according requirement
- Coding techniques
- Testing abilities
- Maintenance section

Requirements Gathering stage:

Gathering a requirement process which accept goals to identify the high level requirements section of the project plan. Every goal will be refined into a set of requirements. All the requirements represent the major functions for the intend application, define

Operational data areas and reference data areas, and define the initial data entities. Major functions include critical processes to be managed, as well as mission critical inputs, outputs and reports. A user class hierarchy is developed and associated with these major functions, data areas, and data entities. Each of these definitions is termed a Requirement. Requirements are identified by unique requirement identifiers and, at minimum, contain a requirement title and textual description.

CPSC8985   Illinois State Finance Corporation
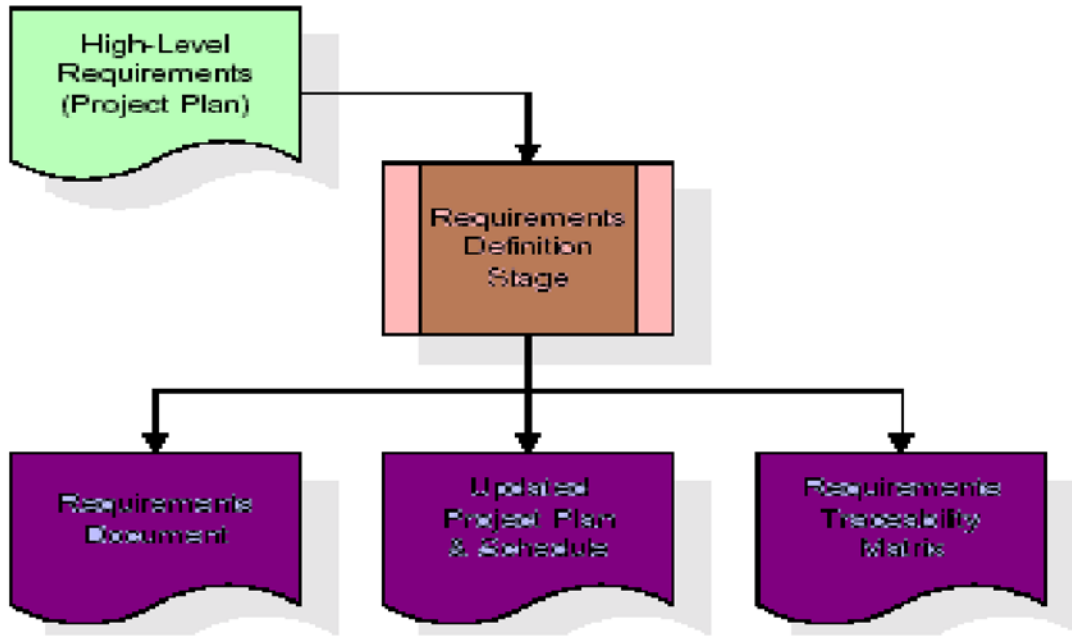
Analysis Stage:

The planning stage establishes a bird's eye view of the intended software product, and uses this to establish the basic project structure, evaluate feasibility and risks associated with the project, and describe appropriate management and technical approaches.

CPSC8985   Illinois State Finance Corporation
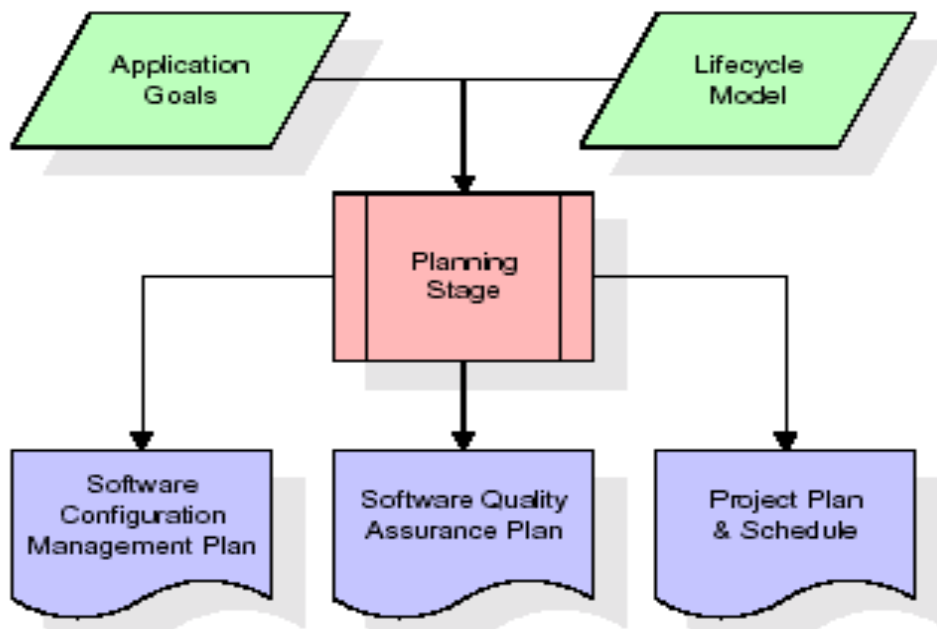
The most basic area of the task arrangement is a maintaning of high state item prerequisites, additionally value to as objectives. The greater part of the product item necessities to be created for the prerequisites definition stage stream from one or a greater amount of these objectives. The base data for every objective comprises of a title and literary portrayal, albeit extra data and references to outer reports might be incorporated. The yields of the undertaking arranging stage are the arrangement administration arrange, the quality confirmation arrangement, and the task plan and timetable, with a point by point posting of booked exercises for the forthcoming Requirements stage, and abnormal state assessments of exertion for the out stages

Designing Stage:

The setup stage takes as its fundamental data the essentials perceived in the attested necessities report. For each essential, a course of action of one or more arrangement parts will be made as a result of gatherings, workshops, and/or model tries. Diagram parts delineate the fancied programming highlights in purpose of interest, and all around fuse useful pecking request charts, screen plan diagrams, tables of business models, business process diagrams, pseudo code, and a complete substance relationship plot with a full data word reference. These setup segments are proposed to delineate the item in sufficient unpretentious component that skilled engineers may develop the item with inconsequential additional information.
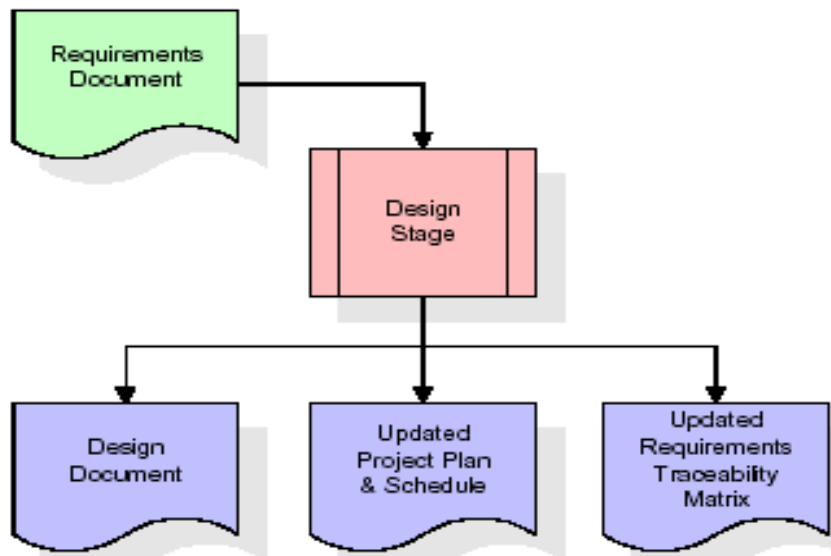
At the point when the outline record is finished and acknowledged, the RTM is overhauled to demonstrate that every configuration component is formally connected with a particular necessity. The yields of the outline stage are the configuration archive, a redesigned RTM, and an overhauled venture arrangement.

Development Area:

The improvement stage takes as its essential information the configuration components portrayed in the endorsed plan archive. For every configuration component, an arrangement of one or more programming ancient rarities will be delivered. Programming curios incorporate however are not restricted to menus, discoursed, information administration shapes, information reporting designs, and concentrated strategies and capacities. Fitting experiments will be created for every arrangement of practically related programming antiques, and an online help framework will be produced to guide clients in their communications with the product.
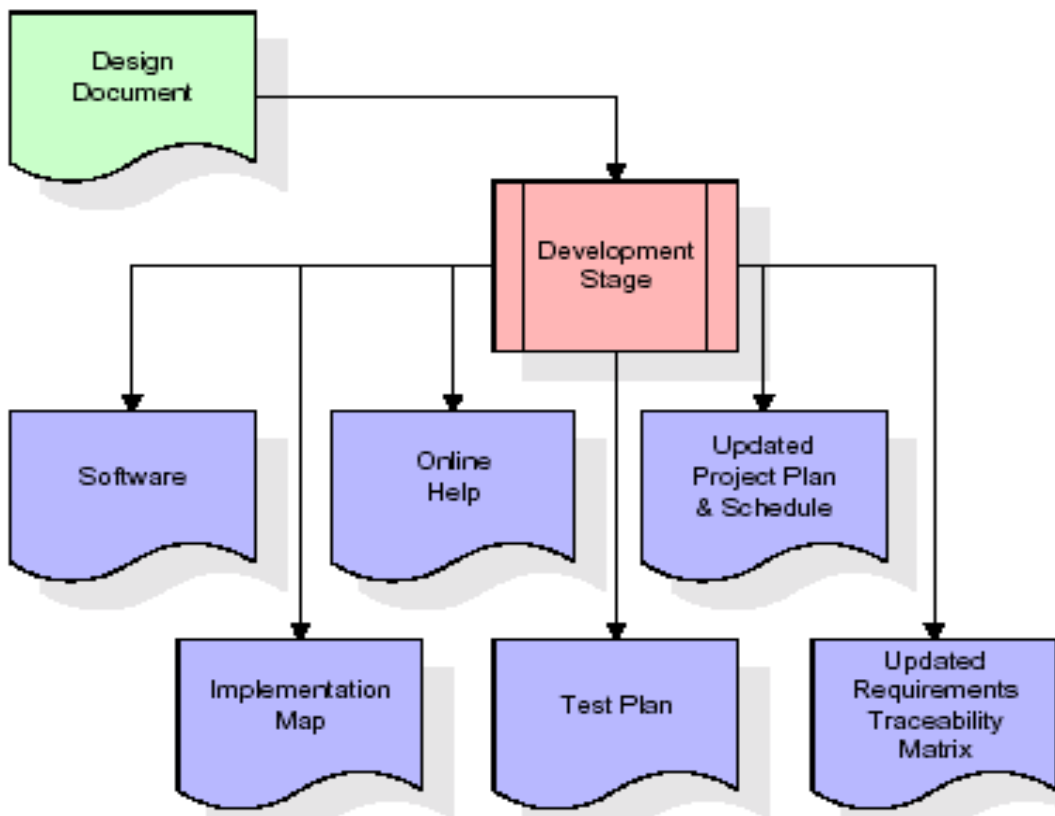

The RTM will be updated to evaluate that each created antiquity is connected to a particular outline component, and that each created ancient rarity has one or all the more comparing experiment things. As of right now, the RTM is in its last design. The yields of the improvement stage incorporate a completely useful arrangement of programming that fulfills the prerequisites and outline components beforehand recorded, an online help framework that portrays the operation of the product, a usage guide that recognizes the essential code section focuses for all real framework works, a test plan that depicts the experiments to

be utilized to approve the accuracy and culmination of the product, an overhauled RTM, and an upgraded venture arrangement.

2.2 Application Information flow:

Below architecture diagram represents mainly flow of requests from users to database through servers. In this scenario overall system is designed in three tires separately using three layers called presentation layer, business logic layer and data link layer. This project was developed using 3-tier architecture.
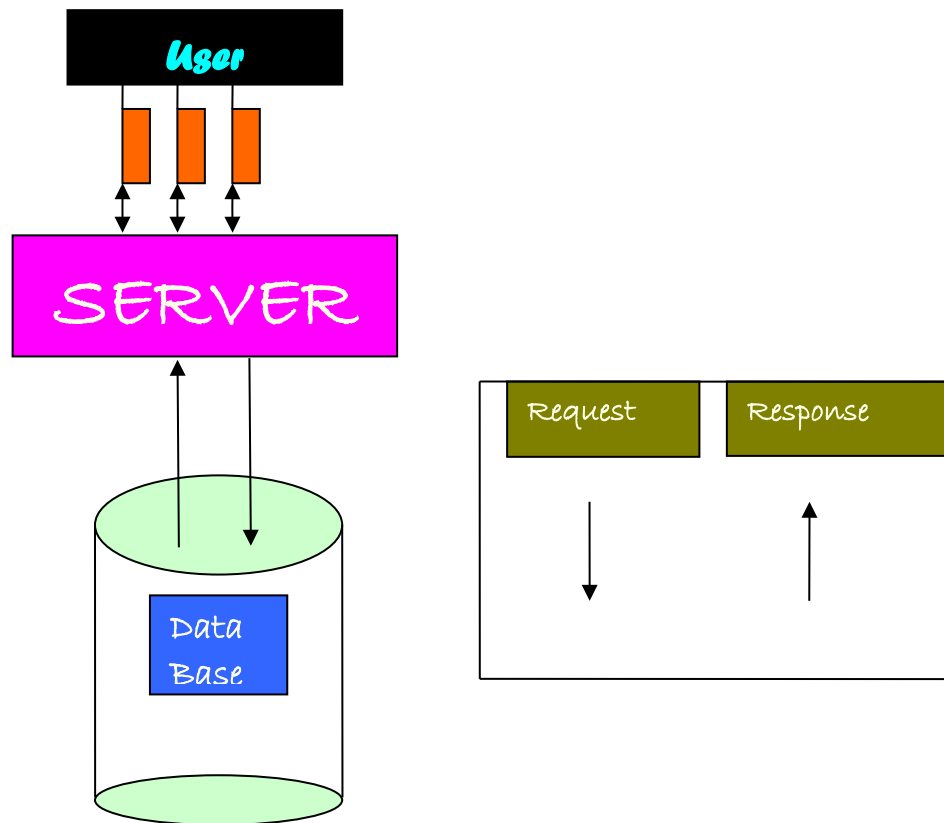
2.3 Interaction with other Projects

In addition to dividing the application into three kinds of components, the Model–view–controller (MVC) design defines the interactions between them.

A controller can send commands to the model to update the model's state (e.g., editing a document). It can also send commands to its associated view to change the view's presentation of the model (e.g., by scrolling through a document).

A model notifies its associated views and controllers when there has been a change in its state. This notification allows the views to produce updated output, and the controllers to change the available set of commands. A passive implementation of MVC omits these notifications, because the application does not require them or the software platform does not support them.

CPSC8985   Illinois State Finance Corporation

2.4 Interactions with other Applications

We have took this project from a reservation system and thought of assigning it to the dorms. Took references form our finance company system. Our project adds a new feature when it come to our loan facility application system. It will give an option to apply loan get response within 48 hours.

2.5 Capabilities

The architecture of this project includes several individual modules. While developing the project each module has developed individually. The following figure shows the process flow of all the modules involved in the project.

2.6 Risk Assessment and Management

Before Risk assessment starts it is basic that an establishment is built up for giving organized venture data, therefore, the accompanying task components were finished and characterized preceding building up this Risk Management Plan:

➢ Define the calendar, assets and expense for the venture.

➢ Able to make a blue print for the Project

➢ Master plan and nitty gritty calendars are to be produced.

➢ Project spending plan ought to be considered.

➢ Required and recognized assets ought to make accessible.

➢ Define Risk Management Roles and Responsibilities

Able to handle the danger administration. Group members has taken a part in assessment of risk gathering and individuals serve as meeting recorders and key partners for danger evaluation sponsors for project may accept the challenges happen in the risk timings.

3   Project Requirements

3.1. Identification of Requirements:

Outlining is the most vital stage. The configuration process includes building up a reasonable perspective of the framework, setting up framework structure, distinguishing information streams and information stores, breaking down abnormal state capacities into sub-capacities, setting up connections, and interconnection among segments and creating solid information representations.

CPSC8985   Illinois State Finance Corporation

With regards to programming, configuration is critical thinking handle whose goal is to Find and depict the best approach to execute the utilitarian necessities while regarding the imperatives forced by the non useful prerequisites and by holding fast to general standards of good quality.

The objective of the outline procedure is to deliver a model or representation of the framework which can be utilized later to fabricate that framework and utilize this model to manufacture the general framework. The outline process makes an interpretation of prerequisites into a representation of the product that can be surveyed for quality before coding starts

3.2. Performance Requirements

The outline handle, the nature of the advancing configuration is surveyed with a progression of formal specialized audits or plan walkthroughs talked about is recommends three qualities that serve as an aide for the assessment of a decent plan.

➢ The outline must actualize the majority of the express necessities contained in the examination model and it must oblige the greater part of the verifiable prerequisites wanted by the client.

➢ The plan must be a discernable, justifiable aide for the individuals who produce code and for the individuals who test and accordingly bolster the product.

➢ The configuration ought to give a complete photo of the product, tending to the information, practical and behavioral spaces from an execution point of view. Each of these attributes is really an objective of the configuration process. Be that as it may, how is each of these objectives accomplished. With a specific end goal to assess the nature of a configuration representation, must set up specialized criteria for good outline.

3.3. Security and Fraud Prevention

Object-oriented design and outline (OOAD) is a product designing approach that models a framework as a gathering of collaborating articles. Every article speaks to some element of enthusiasm for the framework being demonstrated, and is described by its class, its state (information components), and its conduct. Different models can be made to demonstrate the static structure, dynamic conduct, and run-time arrangement of these teaming up items. There are various distinctive documentations for speaking to these models, for example, the Unified Modeling Language (UML).

Object-oriented analysis (OOA) applies object-modeling strategies to dissect the useful prerequisites for a framework. Object-oriented design (OOD) expounds the investigation models to create usage determinations. OOA concentrates on what the framework does, OOD on how the framework does it.

Object-oriented framework is a made out of articles. The conduct of the framework results from the joint effort of those articles. Joint effort between items includes those sending messages to each other. Communicating something specific contrasts from calling a capacity in that when an objective article gets a message, it itself chooses what capacity to complete to administration that message. The same message might be executed by various capacities, the one chose relying upon the condition of the objective item.

The usage of "message sending" shifts relying upon the design of the framework being modeled, and the area of the articles being spoken with.

3.4. Software and Hardware Requirements

An object-oriented system is composed of objects. The behavior of the system is achieved through collaboration between these objects, and the state of the system is the combined state of all the objects in it. Collaboration between objects involves those sending messages to each other. The exact semantics of message sending between objects varies depending on what kind of system is being modeled. In some systems, "sending a message" is the same as "invoking a method". In other systems, "sending a message" might involve sending data via a socket.

3.5. HTML and JavaScript

Object-oriented analysis (OOA) takes a gander at the issue space, with the point of creating a theoretical model of the data that exists in the territory being broke down. Examination models don't consider any usage limitations that may exist, for example, simultaneousness, appropriation, diligence, or how the framework is to be constructed. Usage limitations are managed amid article arranged outline (OOD). Examination is done before the Design.

The hotspots for the investigation can be a composed prerequisites proclamation, a formal vision report, and meetings with partners or other invested individuals. A framework might be isolated into numerous spaces, speaking to various business, mechanical, or different ranges of interest, each of which are investigated independently.

The aftereffect of item situated examination is a portrayal of what the framework is practically required to do, as a reasonable model. That will normally be exhibited as an arrangement of utilization cases, one or more UML class graphs, and various communication outlines. It might likewise incorporate some sort of client interface mock-up.

4. Project Design Description

Loans and estimations has developed easy to access for the customers, where customer no need to special time to spend for these work. This project give us our estimation, appointments, information's for loan types, even customer is able to apply for loans, find the interest rates also.

Project type:

Online Application.

Proposed framework highlights The "INTRANET" is an Enterprise Application that mechanizes and enhances techniques inside associations. It is a valuable innovation for depicting the strides that must be taken, the conditions that must be implemented, and the endorsements that must be acquired amid the fulfillment of ventures.

It is giving simple path to the explorers for picking the better places, lodgings and some more. It is anything but difficult to see the points of interest for the client from the website page. We may run over numerous issues, for example, uncalled for lodging, free of cash amid trasaction.we are giving the bundles less of expense for the voyagers. We give all security tenets to the costumers who are going to visit our travel organization. We give wellbeing, agreeable trip, and quick reservation immediately

We are giving the offices, for example, while booking, for example, (trains, transports, inns) they can book through there credit or check cards so that their will be no free of cash they will be sheltered of there cash whenever. The greater part of travel requires optional income.discreationary pay is cash left over after every financial commitment (nourishment lease and charges) have been paid.

Organization procedures are improved permitting more opportunity for administration of assets. In the event that the client is administrator he can transfer warnings, for example,

For example, the client can seek the specific subtle elements of the travel and they can login to the site page. They can se the point of interest in the website page, for example, train subtle elements, transport subtle elements, lodging points of interest, and so on.

- User can login with his id and password.
- They can view the details and they can register in to the details and they can search for more information from the webpage.
- They can view the pictures of the places and hotels which are need.
- If they have any problem they can contact the tour manager where the contact details are given in the homepage

5. Project Internal/external Interface Impacts and specification

System Requirement Specification

Software Requirements Specification plays an important role in creating quality software solutions. Specification is basically a representation process. Requirements are represented in a manner that ultimately leads to successful software implementation. Requirements may be specified in a variety of ways. However there are some guidelines worth following:

- Representation format and content should be relevant to the problem
- Information contained within the specification should be nested
- Diagrams and other notational forms should be restricted in number and consistent in use.
- Representations should be revisable.

Hardware Requirements

PROCESSOR: Pentium

SPEED: 233 MHz

MONITOR: SAMTRON

HARD DISK: 4.2 MB

RAM: 128 MB

MOUSE: Logitech

KEY BOARD: TVS

Software Requirements

The software requirements specification is produced at the culmination of the analysis task. The function and performance allocated to the software as a part of system engineering are refined by establishing a complete information description, a detailed functional and behavioral description, and indication of performance requirements and design constraints, appropriate validation criteria and other data pertinent to requirements.

An outline of the Software Requirements Specification:

A simplified outline can be given for the framework of the specifications. This is according to the IEEE Standards.

OPERATING PLATFORM: WINDOWS 98/2000NTR OR ABOVE

DBMS: ORACLE 8 OR ABOVE

SOFTWARE: JDK 1.3.1_03,

 WEBSERVER: TOMCAT

BROWSER: INTERNET EXPLORER

FRONT END TOOL: HTML

DOCUMENTATION TOOL: MS – WORD 2000

6. Project Design Units Impact

6.1 Functional Area/Design Unit

UML

The Unified Modeling Language (UML) is a standardized specification language for object modeling. UML is a general-purpose modeling language that includes a graphical notation used to create an abstract model of a system, referred to as a UML model.

Definition UML is a general purpose visual modeling language that is used to

1. specify
2. visualize
3. construct
4. document

The artifacts of the software system.

UML is a language It will provide vocabulary and rules for communications and function on conceptual and physical representation. So it is a modeling language.

UML specifying Specifying means building models which are precise, unambiguous and complete. In particular, the UML address the specification of all the important analysis, design and implementation decisions that must be made in developing and displaying a software intensive system.

UML visualization The UML includes both graphical and textual representation. It makes easy to visualize the system and for better understanding.

UML constructing UML models can be directly connected to a variety of programming languages and it is sufficiently expressive and free from any ambiguity to permit the direct execution of models.

UML documenting UML provides variety of documents in addition to raw executable codes.

The system is designed using on UML. The UML modeling and design is a new way of thinking about problems using models organized around real world concepts. UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software.

Reasons to model

- To communicate the desired structure and behavior of the system

- To visualize and control the system's architecture.

- To better understand the system and expose opportunities for specification and reuse

- To manage risk

Uses of uml

      The UML is expected fundamentally for programming escalated frameworks. It has been utilized successfully for such space as Enterprise data framework, managing an account and money related administrations, information transfers, transportation, safeguard/aviation, and retails, medicinal, hardware, investigative fields, conveyed marry

UML is the business standard "dialect" for portraying, envisioning, and reporting object-arranged frameworks. UML is a gathering of an assortment of graphs for contrasting purposes. Every sort of graph models a specific part of article arranged configuration in a straightforward, visual way.

The UML standard indicates precisely how the charts are to be drawn and what every part in the outline implies. UML is not subject to a specific programming dialect, rather it centers one the essential ideas and thoughts that model a framework. Utilizing UML empowers anybody acquainted with its details to immediately read and comprehend outlines drawn by other individuals. There are UML outline for demonstrating static class connections, dynamic transient communications between items, the utilizations of articles, the particulars of an execution, and the state moves of frameworks.

UML diagram consists of the following features

- **Entities**: These may be classes, objects, users or systems behaviors.
- **Relationship**: Lines that model the relationships between entities in the system.
- **Generalization**: A solid line with an arrow that points to a higher abstraction of the present item.
- **Association**: A solid line that represents that one entity uses another entity as part of its behavior.
- **Dependency**: A dotted line with an arrowhead that shows one entity depends on the behavior of another entity.

Types of UML Diagrams UML defines nine types of Diagrams in which they are divided into two categories. They are Static diagrams and Dynamic diagrams.

1. Class diagram

2. Object diagram

3. Use case diagram

4. Sequence diagram

5. Collaboration diagram

6. State chart diagram

7. Activity diagram

8. Component diagram

9. Deployment diagram

1. Class Diagram:

A Class Diagram demonstrates an arrangement of Classes, interfaces, Collaborations and their connections. Class Diagram models class structure and substance utilizing plan components, for example, classes, bundles and questions. It additionally shows connections, for example, control, legacy, affiliations and others.

2. Object Diagram:

An Object Diagram demonstrates an arrangement of Objects and their connections.

3. Use case Diagram:

An utilization case chart demonstrates the relationship among performers and use cases inside a framework. Use case graphs show components from the utilization case model. The utilization case model speaks to usefulness of a framework or a class as showed to outside performers with the framework.

An utilization case chart is a diagram of on-screen characters, an arrangement of utilization cases encased by a framework limit, correspondence (cooperation) relationship between the performers and the utilization cases, and speculations among the utilization cases.

4. Sequence diagram:

A grouping chart demonstrates an association masterminded in time arrangement. Specifically, it demonstrates the articles partaking in the collaboration by their "helps" and the messages that they trade orchestrated in time succession. It doesn't demonstrate the relationship among the items. A succession graph speaks to an Interaction, which is an arrangement of messages traded among articles inside coordinated effort to impact a coveted operation or result.

5. Collaboration Diagram:

A Collaboration Diagram is an association Diagram that underscores the auxiliary association of the articles that send and get messages. It demonstrates an arrangement of articles, connections among those items and messages sent and got by those items.

6. State Chart Diagram:

A State Chart Diagram demonstrates a state machine, comprising of states, moves, occasions and exercises. State Diagram accentuates the occasion requested conduct of articles which is particularly valuable in displaying responsive frameworks.

7. Activity diagram:

An Activity diagram is a dynamic diagram that shows the activity and the event that causes the object to be in the particular state.

Activity diagrams show the flow of activities through the system. Diagrams are read from top to bottom and have branches and forks to describe conditions and parallel activities. A fork is used when multiple activities are occurring at the same time. The branch describes what activities will take place based on a set of conditions. All branches at some point are followed by a merge to indicate the end of the conditional behavior started by that branch. After the merge all of the parallel activities must be combined by a join before transitioning into the final activity state.

8. Component Diagram:

A Component Diagram shows a set of Components and their relationships. Component diagram are related to class diagram.

9. Deployment Diagram:

A Deployment diagram shows a set of nodes and their relationship. Deployment diagram are related to component diagram.

6.1.1 Functional Overview

 DATA FLOW

REQUEST

ADMIN

WEBSITE

RESPONSE

DATABASE

CPSC8985   Illinois State Finance Corporation

USER   LOGIN   NO   REGISTER

YES

HOME
PAGE

HOME

loan
details

type of
loans

DESTINATION,S
OURCE

select
option

SENDS
INFORMATION   RETURNS
DATA
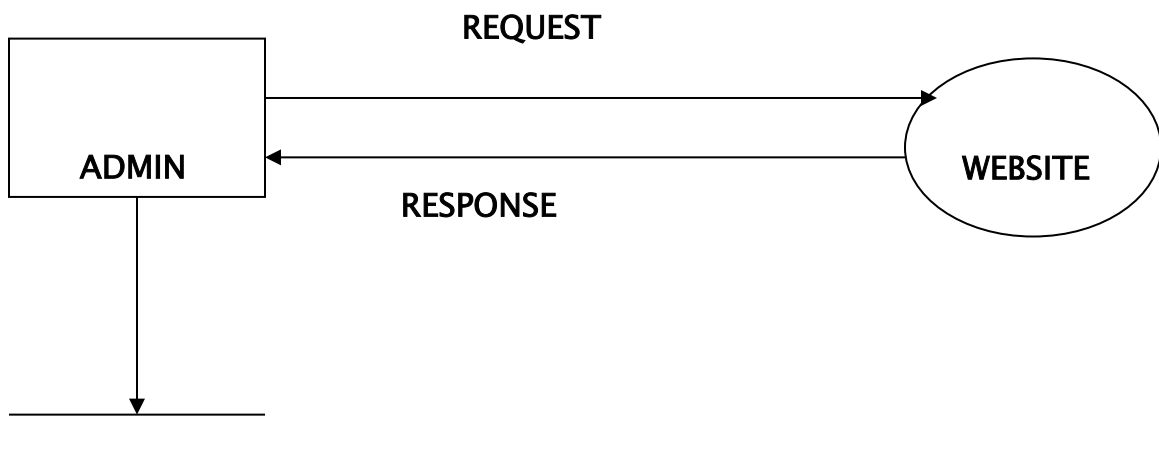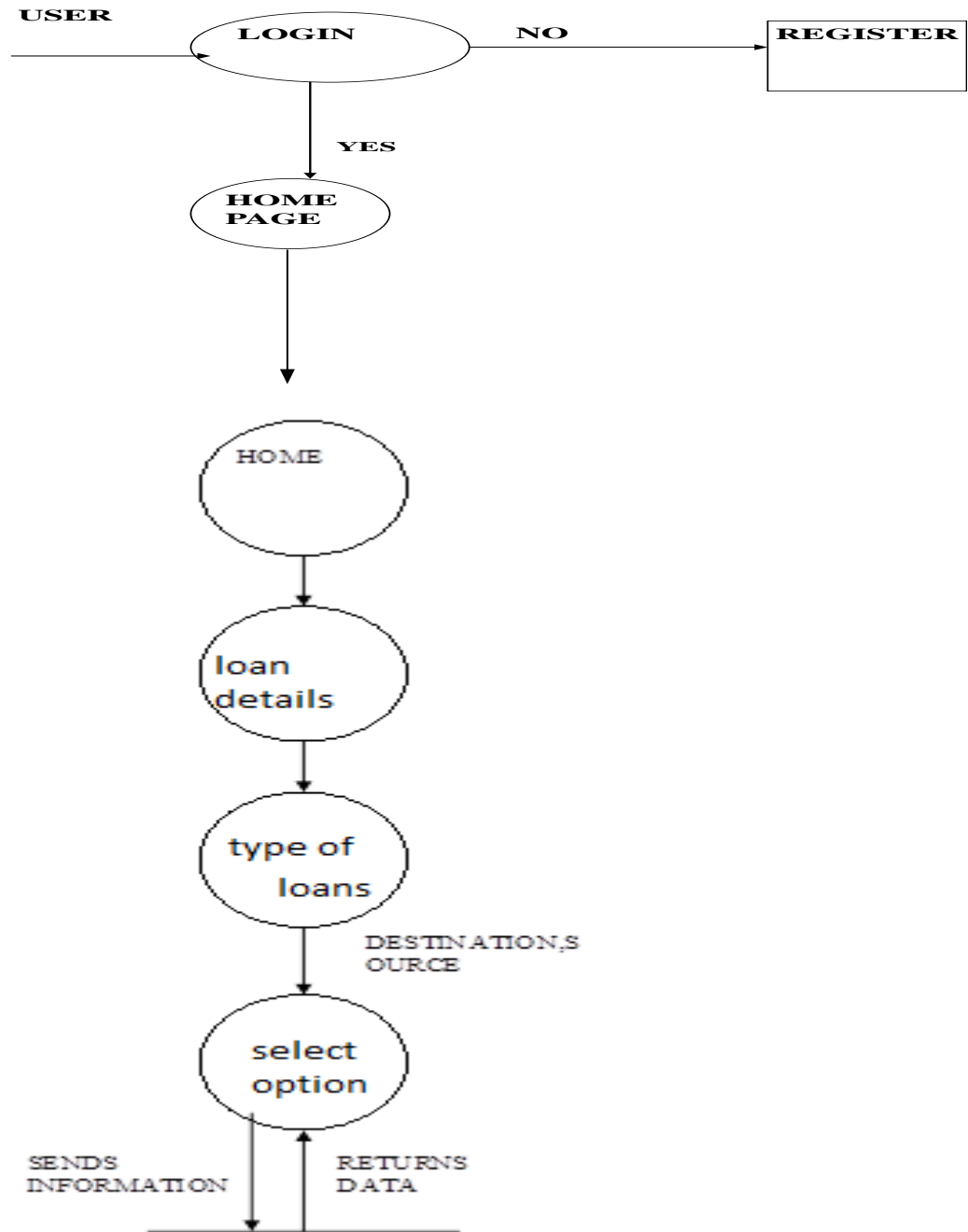
USE CASE DOCUMENTATION

USE CASE DIAGRAM-1

Loan Management

CLASS DIAGRAM-2

CPSC8985   Illinois State Finance Corporation
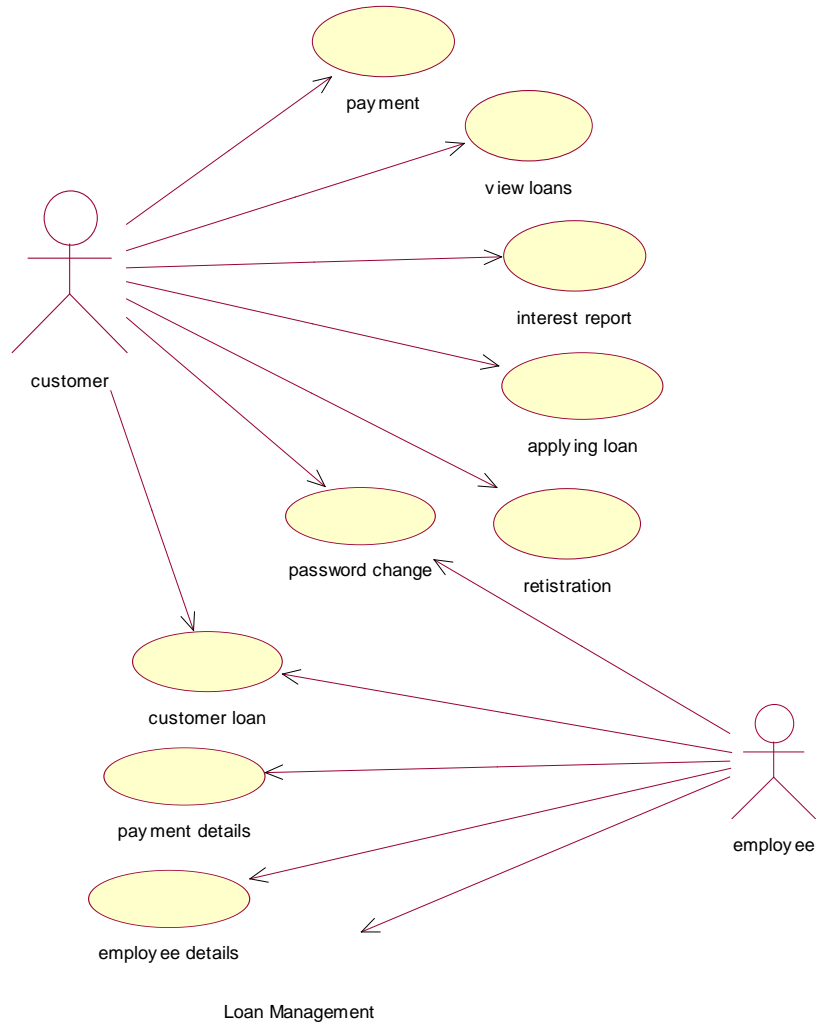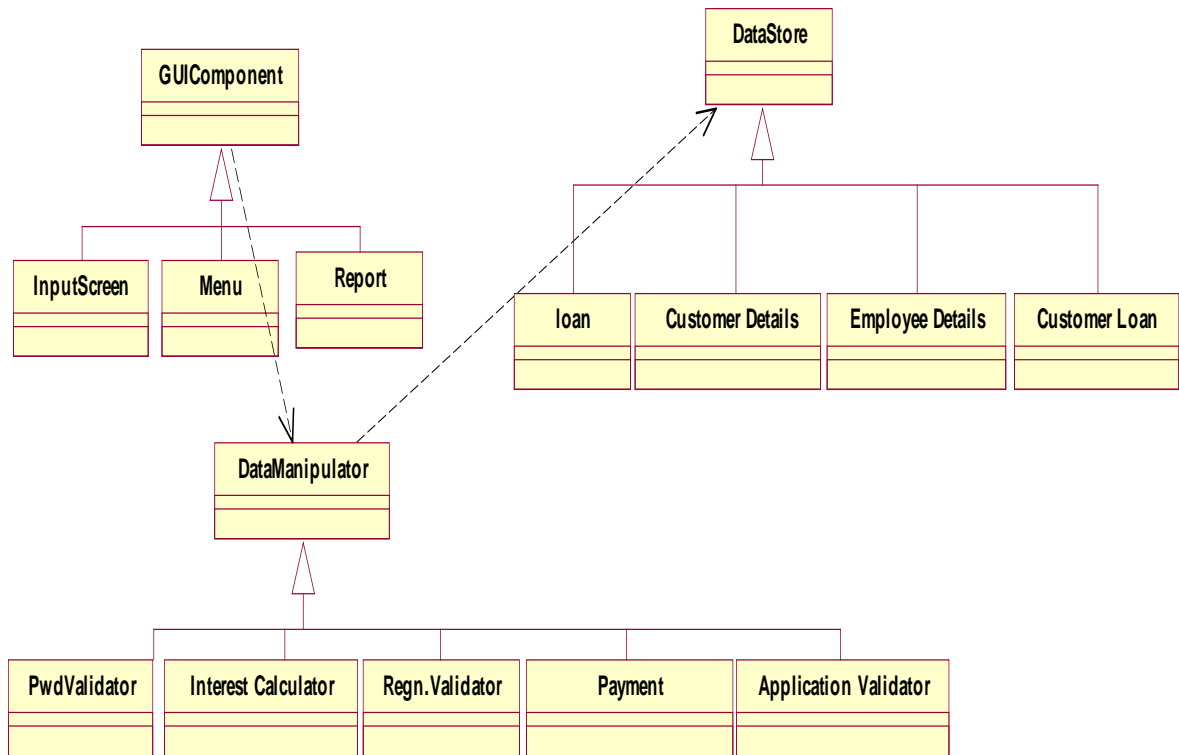
PROCESS FLOW/SYSTEM FLOW CHART /ACTIVITY DIAGRAMS

CPSC8985   Illinois State Finance Corporation
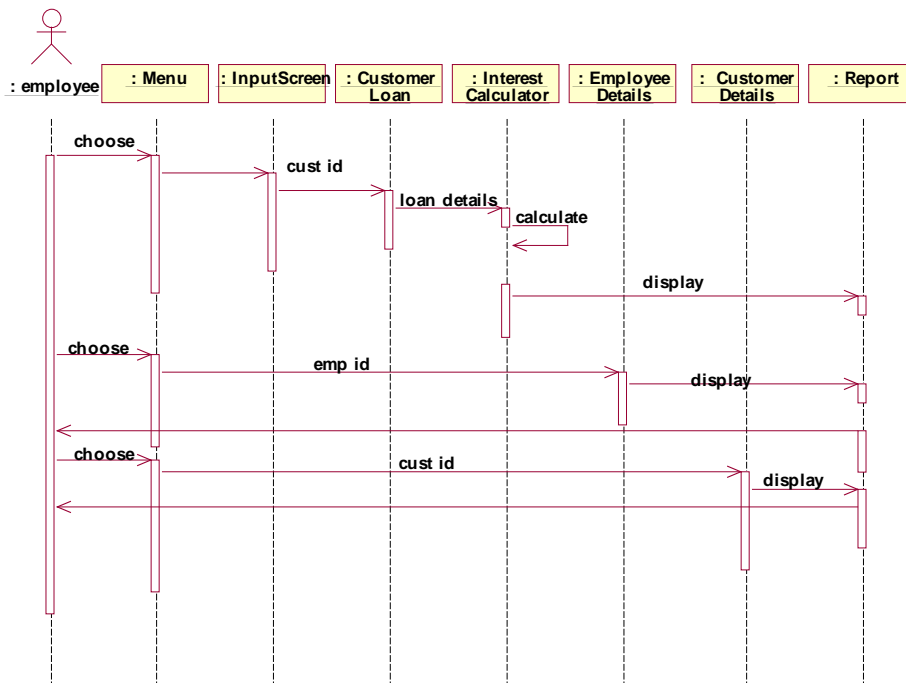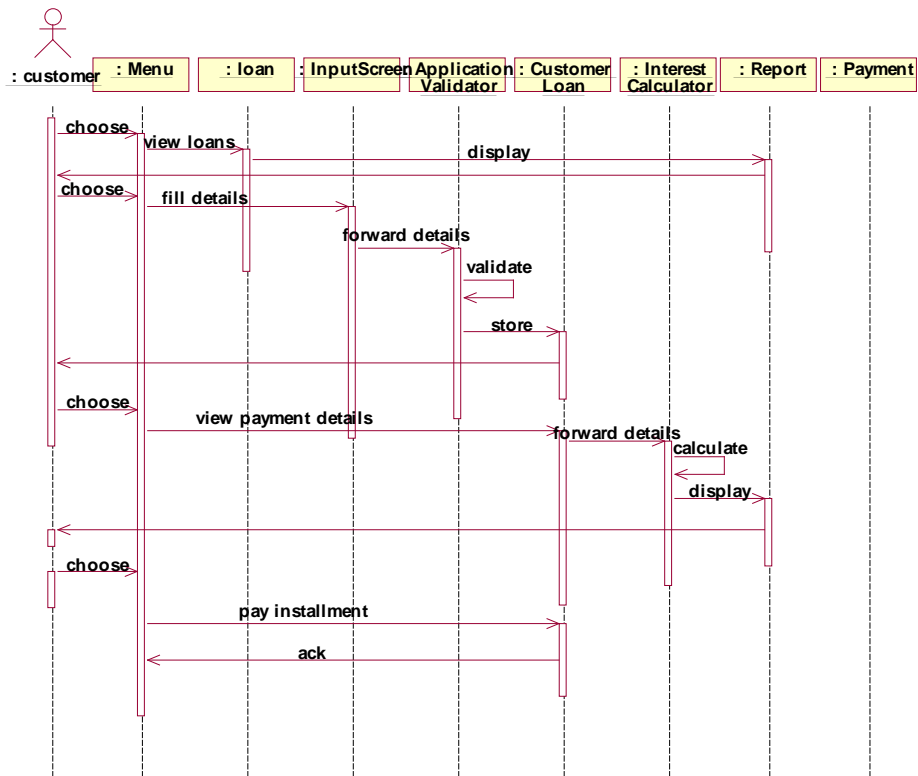
Functional Description

Functional Requirements

Functional Requirements describe the features, behavior, business rules and general functionality that the proposed system must support.

FRF01

If the user is admin he can manages the account access,customer accounts, payments, loans sections. He can view the profiles of online account users.

FRF02

If the user is customer he can apply for loans, verify the payments, able to receive the estimations for desired loan types. The main role for the customers is no need to travel till the office he can easily find his estimation.

Non Functional Requirements

- Usability will be friendly
- Reliability will be supported.
- Performance will be sorted.
- Supportability
- Security is provided.

Usability   The system is designed for the organizations who want to provide security for their information.

Reliability   The system is said to be reliable because the entire system was built using java which is most robust. Reliability refers to the standards of the system.

Performance   The system is highly functional and good in performance. The system must use the minimal set of variables and minimal usage of the control structures to increase the performance of the system dynamically.

Supportability   The system is supportable with different versions of Windows platforms.

Security: The total system is intended to be developed for the security purposes.

6.1.2. Impacts:

By following the above processes we can get to know the complete flow of the processes. By watching the activity diagram we can get to know everything in the project. If there is any issue then we can easily find out by following the UML and Activity diagrams.

6.2 Functional Area/Design Unit B

6.2.1 Functional Overview

6.2.2 Feasibility Report

Preliminary investigation examines project feasibility; the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging old running system. All systems are feasible if they are given unlimited resources and infinite time. There are aspects in the feasibility study portion of the preliminary investigation:

- Technical Feasibility
- Operation Feasibility
- Economic Feasibility

Technical Feasibility

The technical issue usually raised during the feasibility stage of the investigation includes the following:

- Does the necessary solution suggested?
- Do the proposed equipment's have the technical capacity to hold the data required to use the new system?
- Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?
- Can the system will be automatically upgrades?

Are there technical guarantees of accuracy, reliability, ease of access and data security?

Operational Feasibility

User-friendly

Customer will use the structures for their distinctive trades i.e. for including new courses, seeing the courses purposes of interest. Moreover the Customer needs the reports to see the distinctive trades in perspective of the impediments. These structures and reports are made as simple to use to the Client.

The group wills get current trades on line. Regarding old trades, User will enter them into the structure.

Security:

The web server and database server should be protected from hacking, disease et cetera.

Conservativeness:

The application will be delivered using standard open source programming (Except Oracle) like Java, tomcat web server, Internet Explorer Browser et cetera these item will work both on Windows and Linux o/s. In this manner mobility issues won't rise.

Availability

This software will be available always.

Maintainability

The system uses the 2-tier architecture. The 1st tier will be GUI, which called as front-end and the 2nd tier is the database, which uses MYSQL, which is known as back-end.

The front-end can be run on different systems (clients). The database will be running at the server. User will only be access with the id's and passwords.

Economic Feasibility

The framework that deals with the present existing system information stream and techniques totally and control to create all the reports of the manual system other than a large group of other administration reports. It used to be worked as a software application with particular web server and database server. This is required as the exercises are spread all through the association client needs a brought together database. Encourage a portion of the connected exchanges happen in various areas Open source programming like TOMCAT, JAVA, Mysql and Linux is utilized to minimize the expense for the Customer.

6.2.3 System Testing and Implementation:

In order to make sure that the system does not have errors, the different levels of testing strategies that are applied at differing phases of software development are:

Unit Testing:

In this strategy some test cases are generated as input conditions that fully execute all functional requirements for the program. This testing has been uses to find errors in the following categories:

Incorrect or missing functions

Interface errors

Errors in data structure or external database access

Performance errors

Initialization and termination errors.

In this testing only the output is checked for correctness. The logical flow of the data is not checked.

White Box testing:

In this the test cases are generated on the logic of each module by drawing flow graphs of that module and logical decisions are tested on all the cases. It has been uses to generate the test cases in the following cases:

Guarantee that all independent paths have been executed.

Execute all logical decisions on their true and false Sides.

Execute all loops at their boundaries and within their operational bounds

CPSC8985   Illinois State Finance Corporation

Integrating Testing:

Integration testing ensures that software and subsystems work together a whole. It tests the interface of all the modules to make sure that the modules behave properly when integrated together.

System Testing:

Involve in-house testing of the entire system before delivery to the user. Its aim is to satisfy the user the system meets all requirements of the client's specifications.

Acceptance Testing:

It is a pre-delivery testing in which entire system is tested at client's site on real world data to find errors.

Test Approach:

Testing can be done in two ways:

Bottom up approach

Top down approach

Bottom up Approach:

Testing can be performed beginning from littlest and most reduced level modules and continuing each one in turn. For every module in base up testing a short program executes the module and gives the required information so that the module is requested that perform the way it will when installed inside the bigger framework. At the point when base level modules are tried consideration swings to those on the following level that utilization the lower level ones they are tried exclusively and after that connected with the beforehand analyzed lower level modules.

Top down approach:

This kind of testing begins from upper level modules. Since the point by point exercises generally performed in the lower level schedules are not gave stubs are composed. A stub is a module shell called by upper level module and that when achieved legitimately will give back a message to the calling module demonstrating that appropriate connection happened. No endeavor is made to check the accuracy of the lower level module.

6.3 OUTPUT SCREENS:

CPSC8985   Illinois State Finance Corporation

Work Flow:
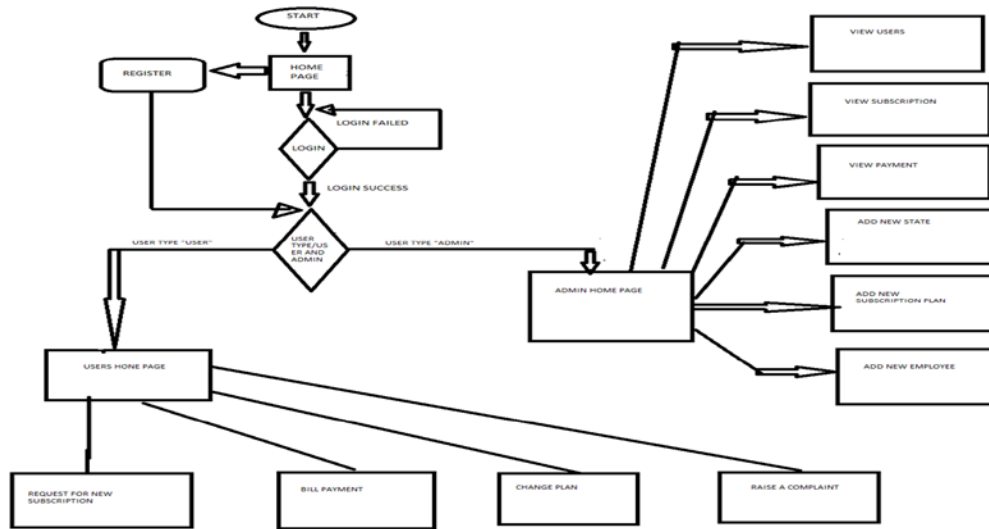
DATABASE:

TABLE FOR LOGIN:
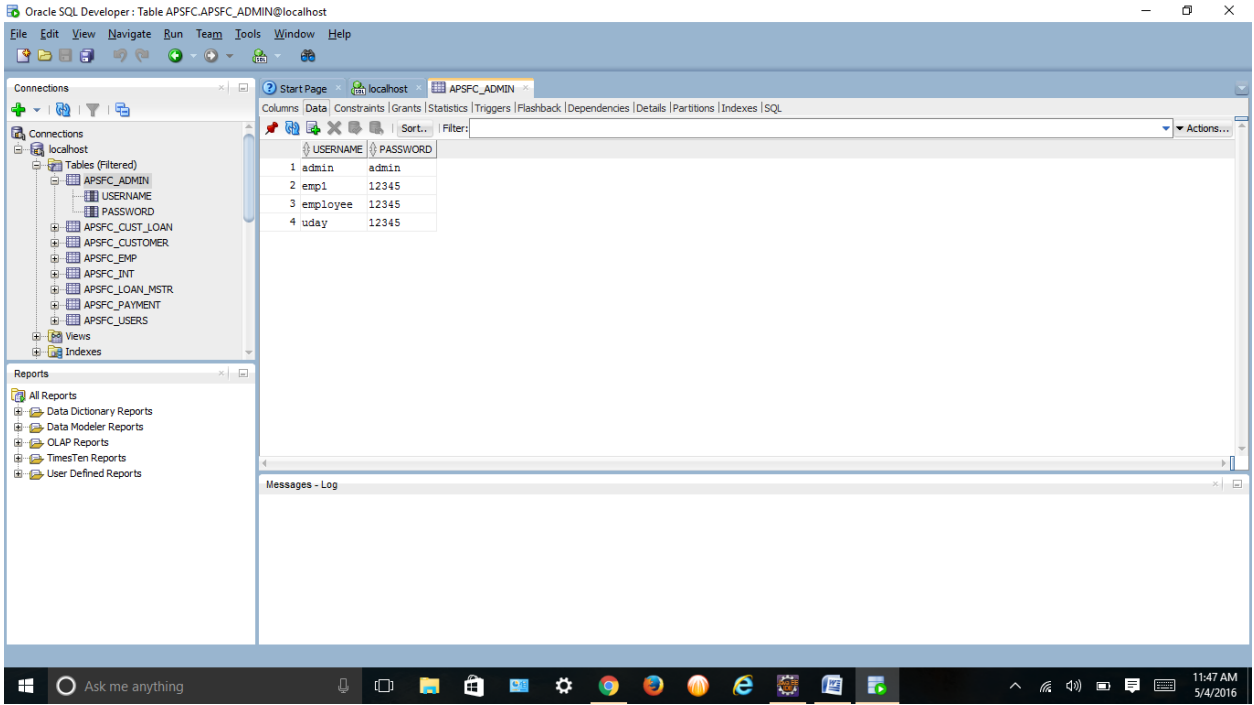
CUSTOMER DETAILS TABLE:

EMPLOYEE DETAILS:

CPSC8985   Illinois State Finance Corporation
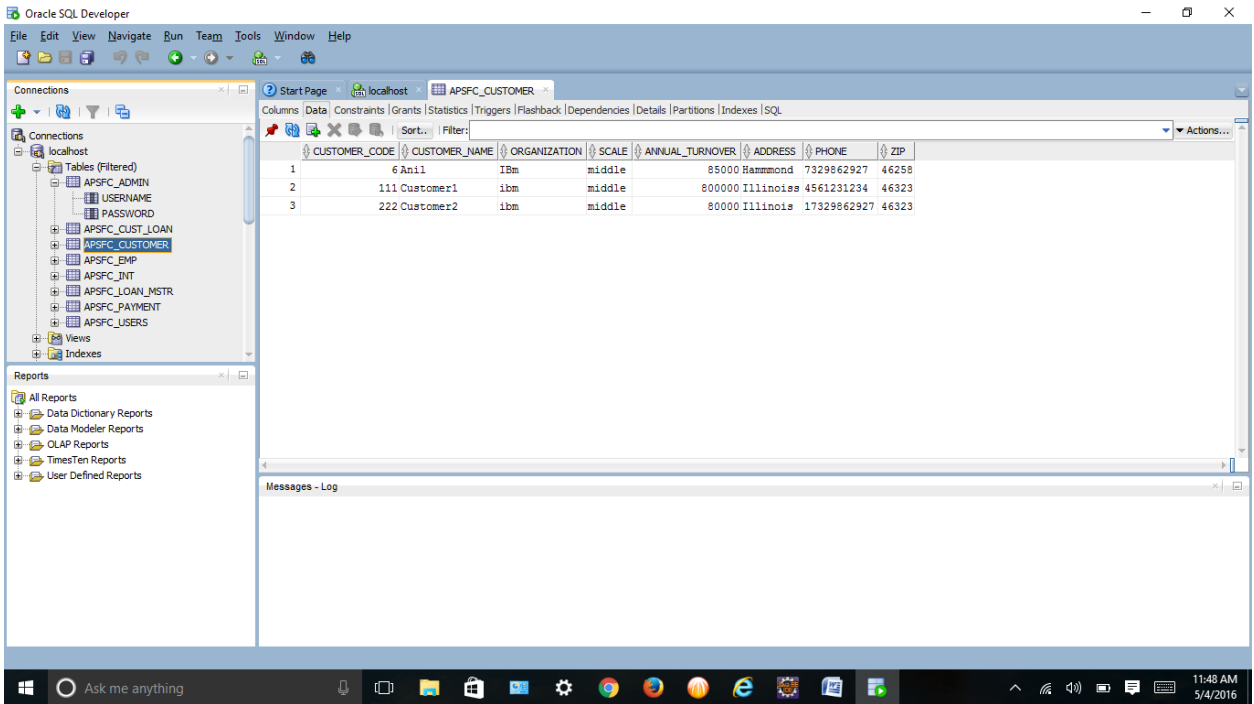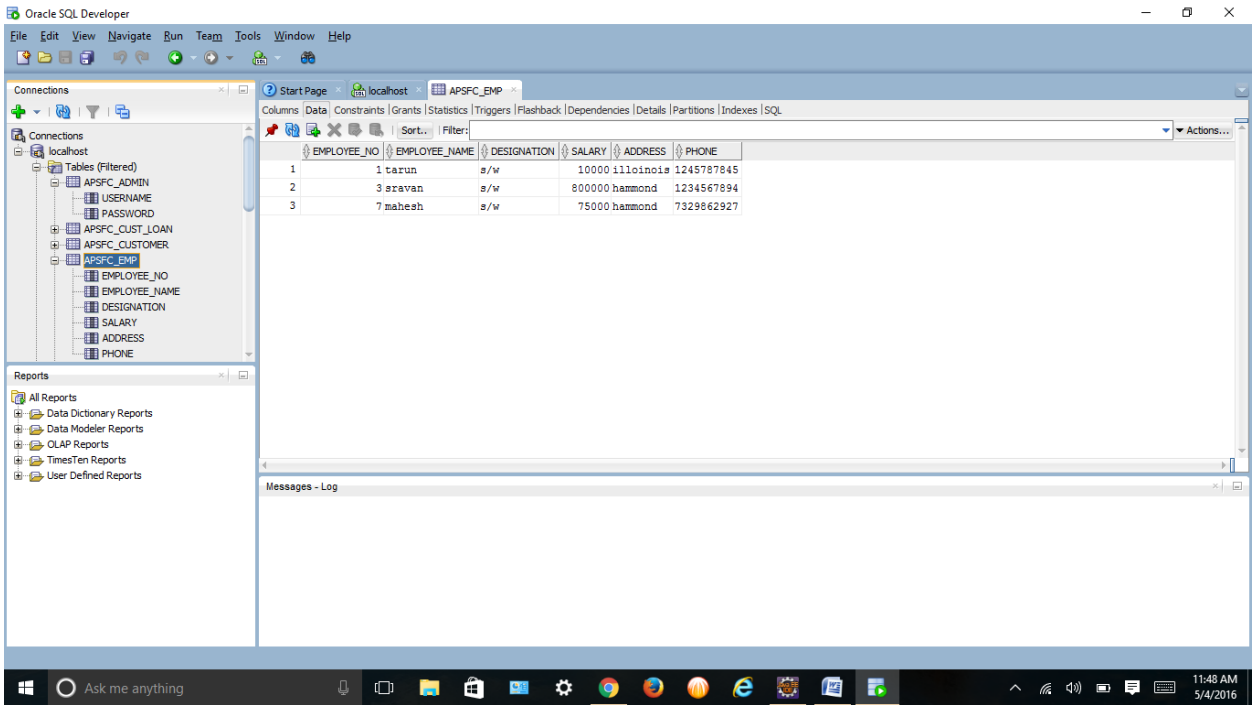
INTEREST TABLE:
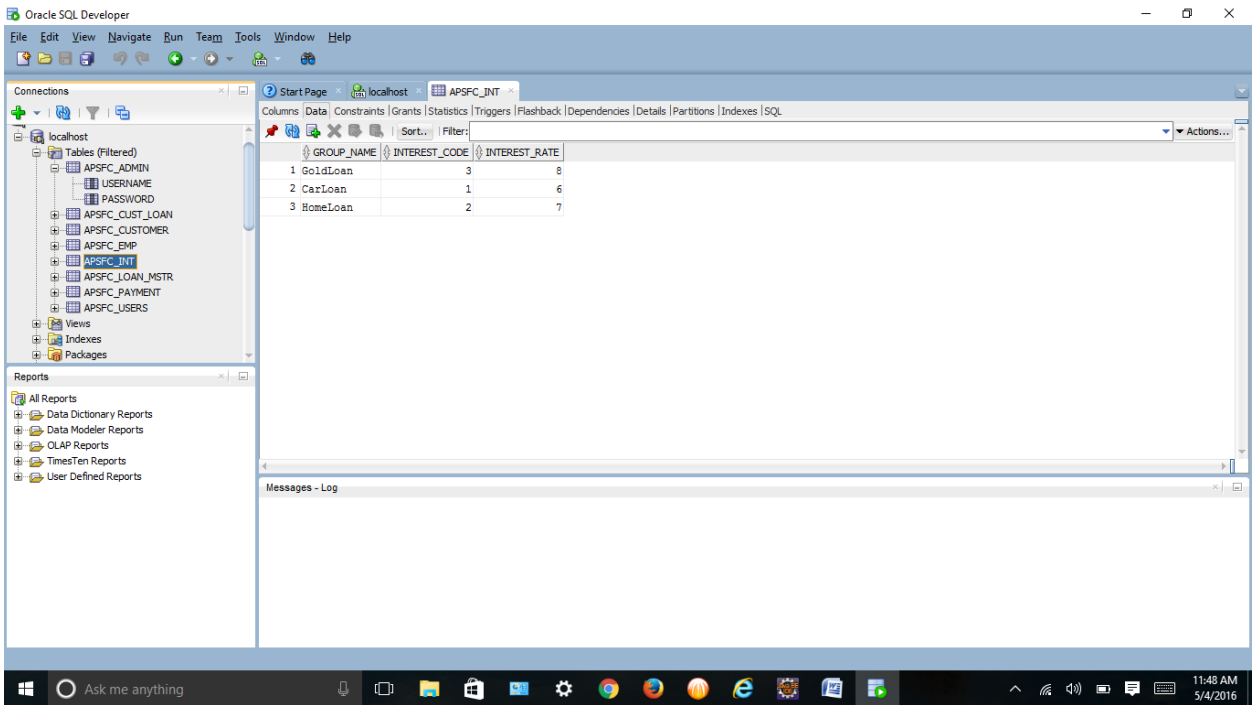
USERS DETAILS STORAGE SPACE:

CPSC8985   Illinois State Finance Corporation

Home Page:

CPSC8985   Illinois State Finance Corporation
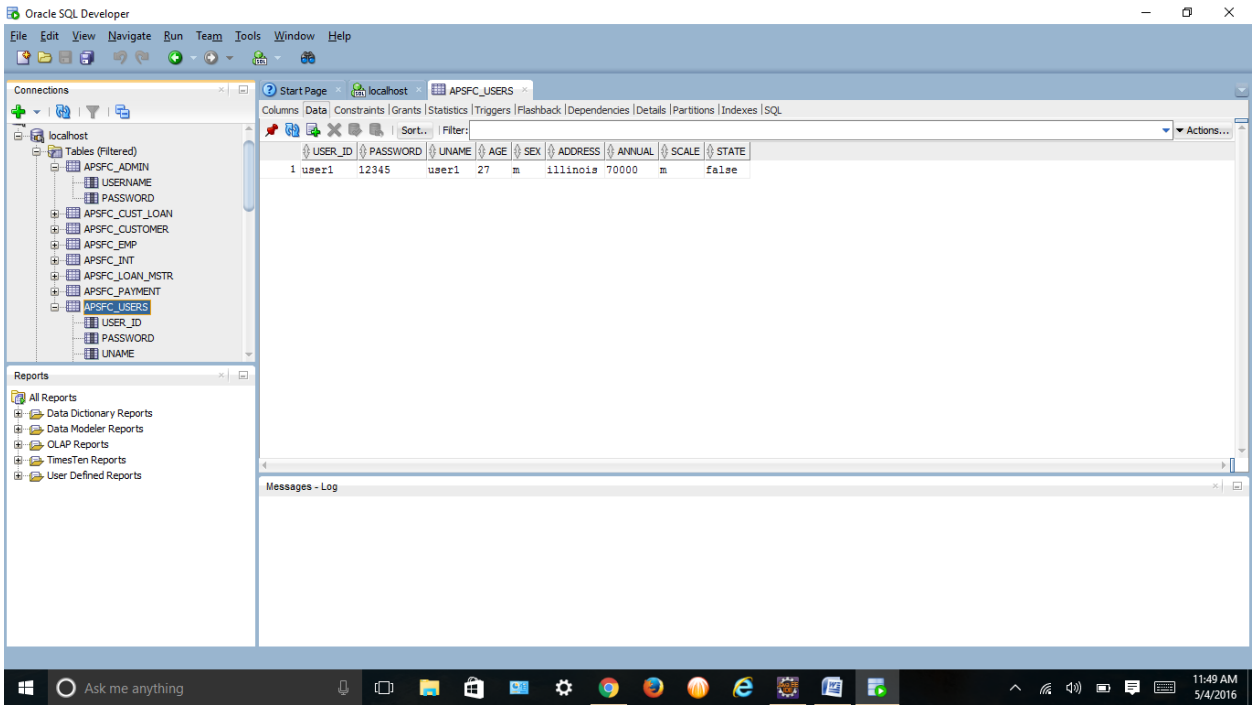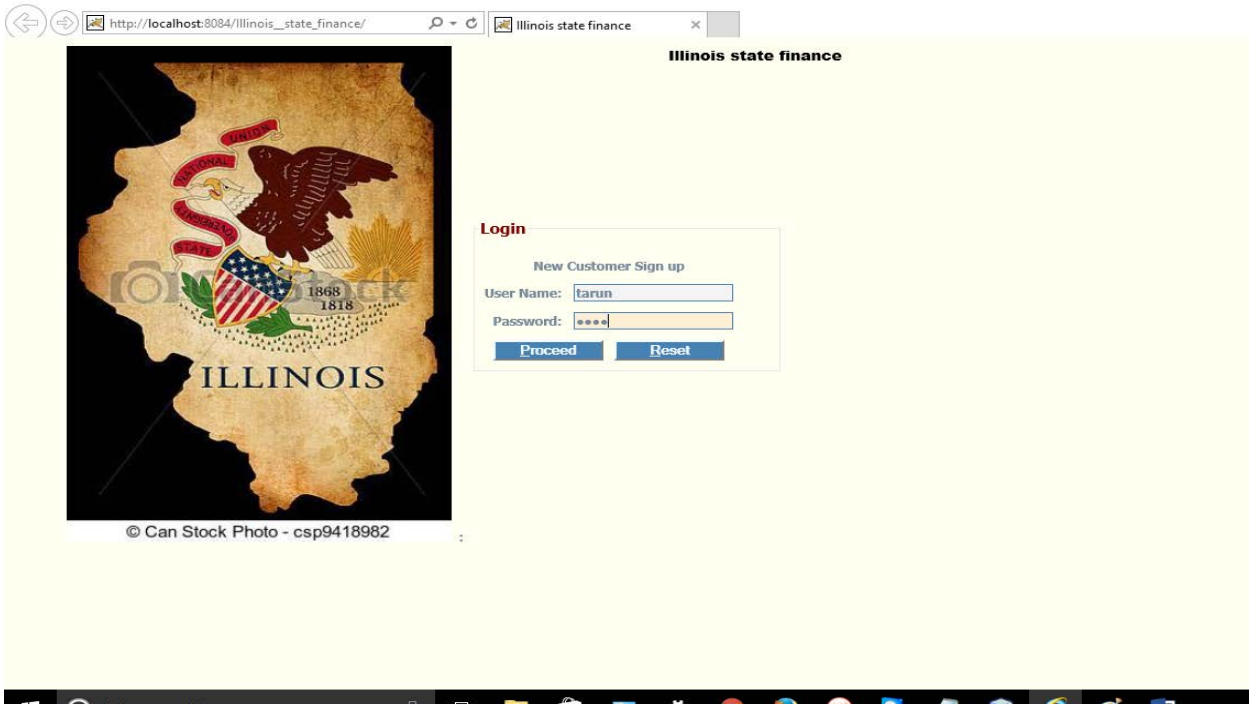
MENU PAGE:

LOAN DETAILS PAGE:

CPSC8985   Illinois State Finance Corporation

INTEREST RATE VALUE:

LOAN CODE DECLARATION:

CPSC8985   Illinois State Finance Corporation

HOME MENU FOR ADMIN:

CPSC8985   Illinois State Finance Corporation

CUSTOMER DETAILS:

INTEREST SELCTION:

PAYMENT ACCESS FOR CUSTOMER:

PRINT/PRIVIEW FOR CUSTOMER REGARDING PAYMENTS:

userid	password	Name	Age	Sex	Address	Anual Income	Scale
nikil	nikil	nikil	22	m	hyd	100000	m
tarun	2328	tarun	23	m	44545	5000000	h

CPSC8985   Illinois State Finance Corporation

CPSC8985   Illinois State Finance Corporation

CPSC8985   Illinois State Finance Corporation

CPSC8985   Illinois State Finance Corporation

Transaction Terminal

Loan code	Customer Code	Imterest Code
789	111	1
555	222	2
56789	111	3

Home   Close

Customer Payment Information

User Profile
Settings
Logout

Loan No   select ▾

Payment Amount

Payment Date   Jan ▾   2016 ▾

Credit Card No

Save It   Refresh

Home   Exit

CPSC8985   Illinois State Finance Corporation

REPORTS GENERATION PAGES:

Print The Reports

CUSTOMER-CODE	CUSTOMER-NAME	ORGANIZATION	SCALE	ANNUAL-TURNOVER	ADDRESS	PHONE	ZIP
6	Anil	IBm	middle	85000	Hammmond	7329862927	46258
111	Customer1	ibm	middle	800000	Illinoiss	4561231234	46323
222	Customer2	ibm	middle	80000	Illinois	17329862927	46323

Print The Reports

EMPLOYEE-NO	EMPLOYEE-NAME	DESIGNATION	SALARY	ADDRESS	PHONE
1	tarun	s/w	10000	illoinois	1245787845
3	sravan	s/w	800000	hammond	1234567894
7	mahesh	s/w	75000	hammond	7329862927

CPSC8985   Illinois State Finance Corporation

Print The Reports

GROUP-NAME	INTEREST-CODE	INTEREST-RATE
GoldLoan	3	8
CarLoan	1	6
HomeLoan	2	7

7. Open Issues

- Technical –problem in the project.
- Business process –project's design process.
- Change management – Any changes for business, customer, or environmental changes.
- Resource – supporting equipment, material, or people problems.
- Third party –Issues from third party with vendors, suppliers, or another outside party.

8. ACKNOLEDGEMENTS

We drive our great pleasure in expressing our sincere gratitude to advisor Dr.Soon Oak Park and Professor Do-yung Park for their timely suggestions, which helped us to complete the project work successfully.

9 REFERENCES:

- Database System Concepts sixth edition by Abraham silbershatz, Henry F. Korth, S. Sudarshan.
- J. Han, J Pei and M. Kamber, "Data mining: concepts and techniques'" (3nd ed.). Morgan Kaufmann Publishers, 2006.
- H. Edelstein, Introduction to Data Access and Knowledge Discovery, Third Edition, ISBN: 1892095025, Two Crows Corporation, 1999.

- www.w3schools.com.
- http://www.apachefriends.org

10 Appendices