Bits ID: 2019ht13096
Name: Braj Kishor
Course No: S2-19_SSZG653
Course Title: Software Architectures

Q.1 Solution

Following are the requirements of the proposed system:

- Functional Requirements:
    - Ability for customers to access and apply for loan using form-based applications and upload the relevant documents.
    - Ability for Loan officers to list and see all the applications applied with filtering and querying capabilities.
    - Ability for the Loan officers to interact with the customer (via Email) for requesting additional details, documents and approvals.
    - Ability for Loan officer to get the CIBIL score of the customer and calculate the maximum eligible amount of loan for the customer.
    - A process/workflow flow-based design where the loan application can be moved through stages and it can be approved or rejected at each stage.
    - Ability for the Finance officer to Disburse the loan to customer account using the banking system.
    - Ability for the customer to track the status of their loan application.
    - Ability to customise or add new steps in the solution.

- Non-Functional Requirements:
    - **Security**: Since this system deals with finance and liability, the system should have very high degree of security characteristics, especially in the following areas
        - **Authenticity & Non-Repudiation**: The documents shared by the customer, the loans term & conditions, the final agreement between bank and customer, all of these documents are highly valuable in the process. The process of sharing and agreement of such documents should have proper authenticity and Non-repudiation mechanism so that both parties can trust the documents and neither parties are cheated.
        - **Confidentiality**: The application will contain sensitive financial and social details/documents of the customer. The CIBIL score is also a confidential data. While storing such data, confidentiality of all this data is a must.
        - **Integrity**: The integrity of the loan's details and documents of the loan is highly valuable for the Bank as well as customer. Hence the data stored should have high integrity.
        - **Authorization**: An authorization mechanism should be in place so that the various actors in the system can access only the views designed for them. For ex A Loan Officer should only have ability to view and approve loan, where a Finance officer should have the ability to disburse amount for approved loan.

- **Usability**: Following usability requirements must be met:
  - **Ease of use for Customer**: The system should have an interactive and easy to understand interface for the customer to fill the application form.
  - **Easy to Use for Bank Employee**: On the bank employee side, since the officers will be processing multiple applications each day, the system should have easy to use interface with advanced filtering, sorting and searching capability so that they can easily process and manage the all the applications received. They should also be able to move the application through various stage of the process easily.
  - **Learnability for Bank Employee**: The system's interface should be simple and easily learnable for the bank employee so that they can be productive asap.

- **Modifiability**: The system should have a process flow-based design where additional steps can be added at any point of time.
- **Availability**: The system should be highly available, at least at the bank employee side, so that the accumulated backlog of the loans application does not increase.
- **Interoperability**: The system should be able to interact with different external applications like the CIBIL services API, Client Email Services and the Banking System.
- **Scalability**: The systems will be receiving and processing Lakhs of application per day. Hence it must be able to scale itself to the requirements.

Q.2 Solution

Quality attribute	Architecturally significant requirement	Business value	Architecture impact
Security	While storing the details shared by the customers and bank regarding a loan, Confidentiality and Integrity should be very high.	High	High
Interoperability	The system should be able to interact with different external applications like the CIBIL services API, Client Email systems and the Banking System.	High	High
Usability	The system should be Easy to Learn and Use for the Bank Employee especially in the aspects of managing and processing many loans in single day and moving them across various stages of the process.	High	High
Modifiability	The system should have a process flow-based design where additional steps can be added at any point of time.	High	Medium
Security	Authenticity & Non-Repudiation of all the communication and document shared between bank and customer must be maintained.	High	Medium

Justification of each scenario for choosing them to be in ASR is as follows:

1. ***While storing the details shared by the customers and bank regarding a loan, Confidentiality and Integrity should be very high.***

   The data shared and stored by customer and bank for such a financial transaction will be highly sensitive. The details include loan details (amount, interest), mortgage details, guarantee details, repayment details, terms and conditions, proofs submitted by customer etc.

   Such details can be used by either party for agreements. The same will be used by the bank & customer, in case of, non-payments, defaults, fraud etc. Hence the confidentiality and integrity of such data is highly important.

   Storing such data in a secured way to protect against unauthorized access and tampering, audit trailing of the data etc will require significant architectural consideration.

2. ***The system should be able to interact with different external applications like the CIBIL services API, Email Systems and the Banking System.***

   To get the CIBIL score, the system needs to connect to CIBIL service API. To interact with Banking system for the transfer of funds, the system needs to interface with banking system. The system must also be trigger emails to customer and other stakeholders during various steps for any requirements. These steps are must for processing the loan online.

   All these requirements transfer highly sensitive data. Additionally, these systems will have their own parameter requirements for interfacing. Hence, a robust, generic and highly secure solution of interfacing is needed while considering the architecture.

3. ***The system should be Easy to Learn and Use for the Bank Employee especially in the aspects of managing and processing many loans in single day and moving them across various stages of the process.***

   Bank Employees will be processing multiple loans per day and more loans will keep getting added in backlog. As such, the interface should have enough options and actions, that an employee can easily view, filter, sort, process, get additional details and approve each application carefully. All these options must be easy to access and use. Furthermore, the learning curve of such system should not be high.
   This will ensure that the employee is productive and his decisions are accurate.

   Such a consideration while designing the architecture of the interface is a must.

4. ***The system should have a process flow-based design where additional steps can be added at any point of time.***

   The system demands for a Business Process Management or Workflow Engine like application to ensure that there are various stages with approval and rejections. As such, it should be easy enough to add new stages in this flow or customize any of existing stage.
   Choosing a proper BPMN flow and solutioning it to suit current requirements while keeping it customisable for future is a must.

5. ***Authenticity & Non-Repudiation of all the communication and document shared between bank and customer must be maintained****.*

   There will be multiple documents which will be shared between customer and bank. The transfer media can be upload by customer on website, Email etc. During this transfer, both the parties should be able to trust the documents. Furthermore, once document is verified and agreed upon, no party can repudiate the document. Such documents can be used in legal proceedings.

   The architecture used to transfer, upload and store these documents must have the above point under consideration.

Q.3 Solution

Tactics used to address the top 2 ASR

ASR 1: *Confidentiality and Integrity of the details shared by the customers and bank regarding a loan while storing should be very high.*

Tactics Used:

- **Encryption of the Storage**: The DB and File System on which this data will be stored should be encrypted to ensure the data remains confidential.
- **Gated/Restricted Access and Audit trails**: The access to such storage via any media should be heavily restricted. Read/Write permissions should be controlled. Proper authentication and authorization mechanism should be in place. An audit trail of who accessed and modified what must be maintained at all times. This will ensure no unauthorized access and integrity of data
- **Backups**: The data should have alternative backups and mismatch flagging mechanisms to ensure correctness of data.

ASR 2: *The system should be able to interact with different external applications like the CIBIL services API and the Banking System.*

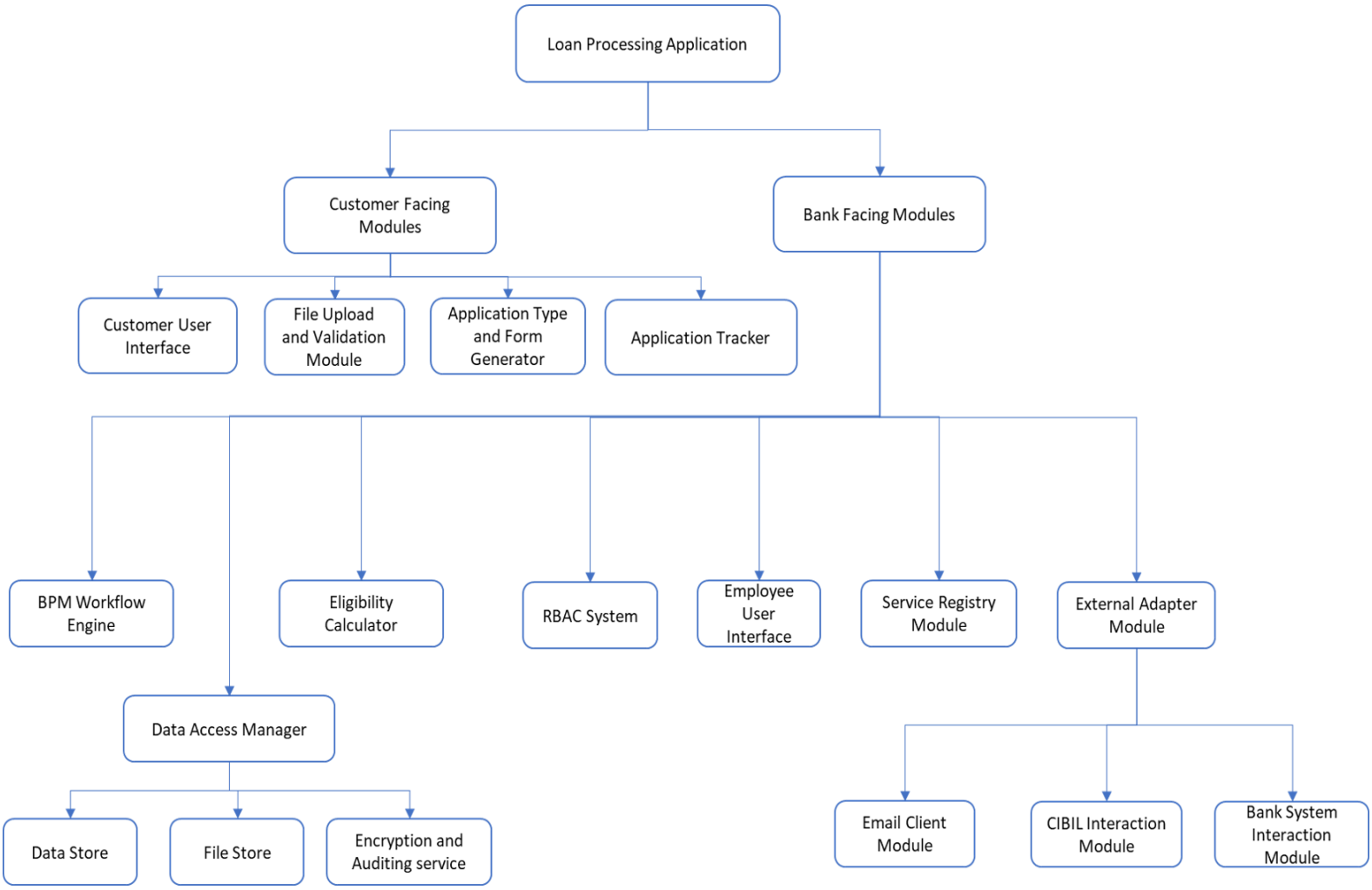
Tactics Used:

- **Service Oriented Architecture:** SOA design consideration should be used so that easy interfacing is available with various external applications. The system should be able to use **REST/SOAP** services, Service discovery mechanisms, on-demand trigger, XML or JSON exchange over HTTPS etc.
- **Service Discovery Mechanisms:** A Service Discovery module can be used as central registry to find and connect with the various external modules.
- **BPM based Workflow Engine:** A BPM based workflow engine can allow using multiple modules at different stages, each of which can be designed to do specific task. Some of them can be specialised for consuming external services on demand.

Bits ID: 2019ht13096
Name: Braj Kishor
Course No: S2-19_SSZG653
Course Title: Software Architectures

Q.4 Solution

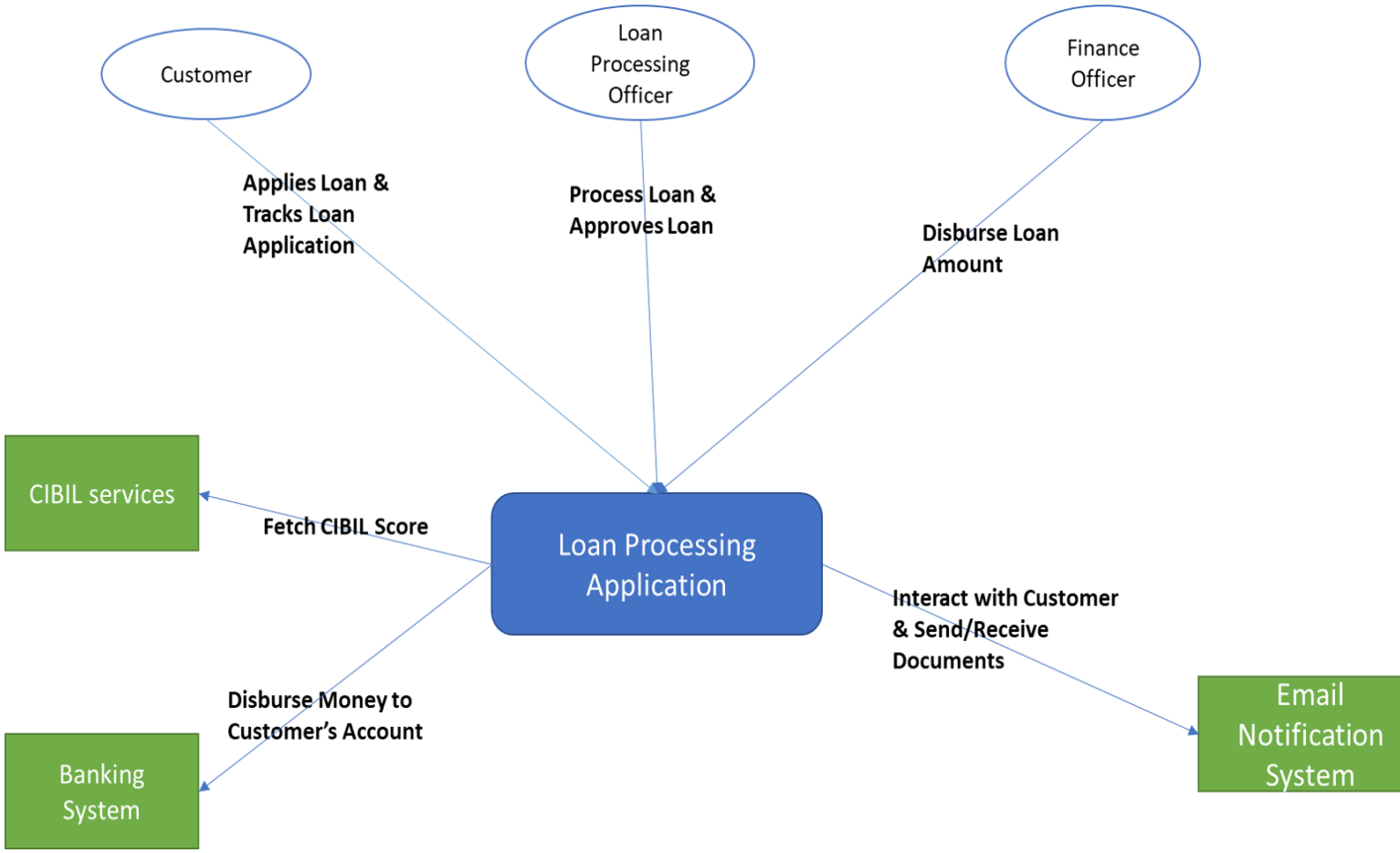
Various Modules of the system:



Bits ID: 2019ht13096
Name: Braj Kishor
Course No: S2-19_SSZG653
Course Title: Software Architectures

Q.5: Solution

Context diagram

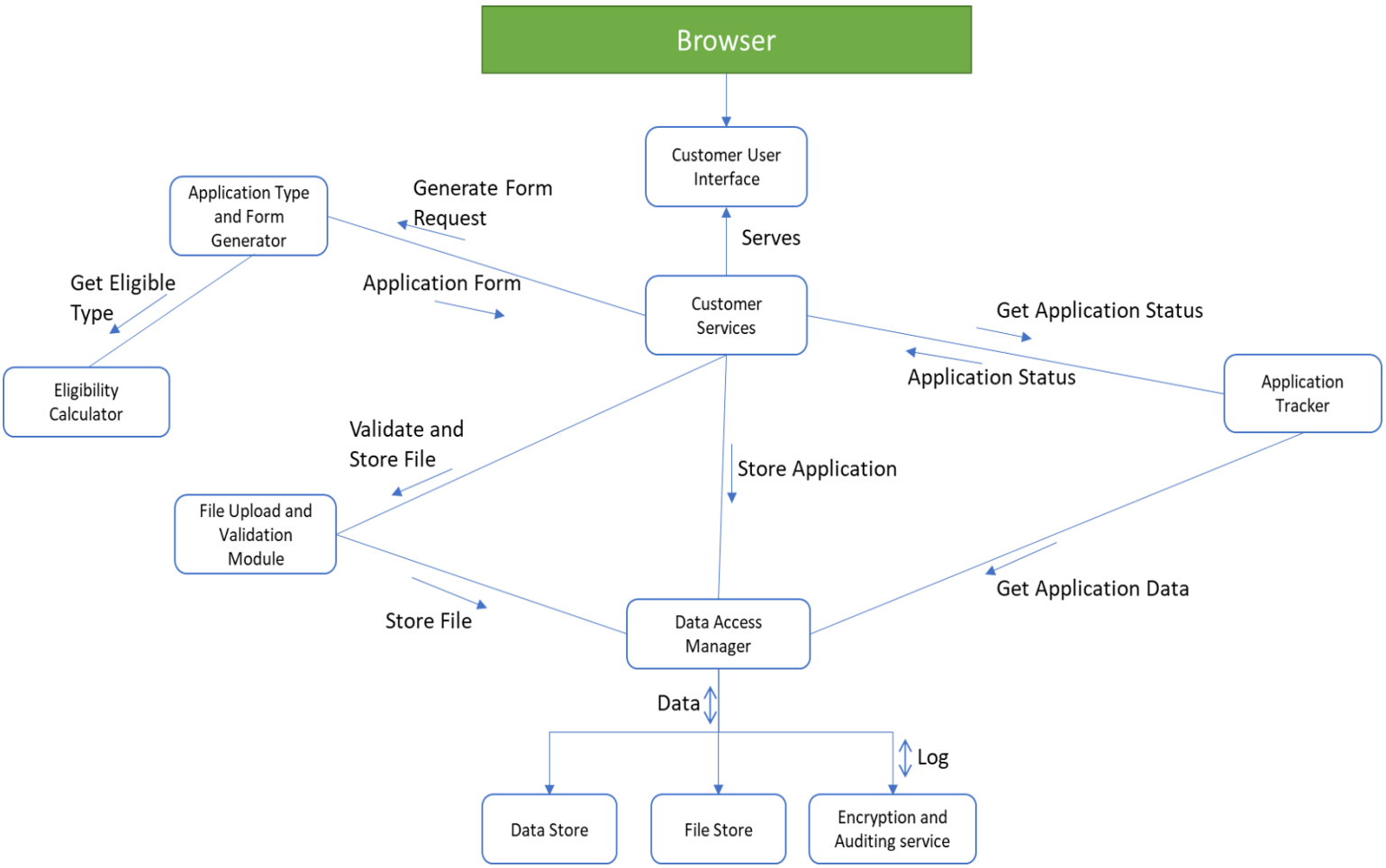


Bits ID: 2019ht13096
Name: Braj Kishor
Course No: S2-19_SSZG653
Course Title: Software Architectures

Q.6 Solution

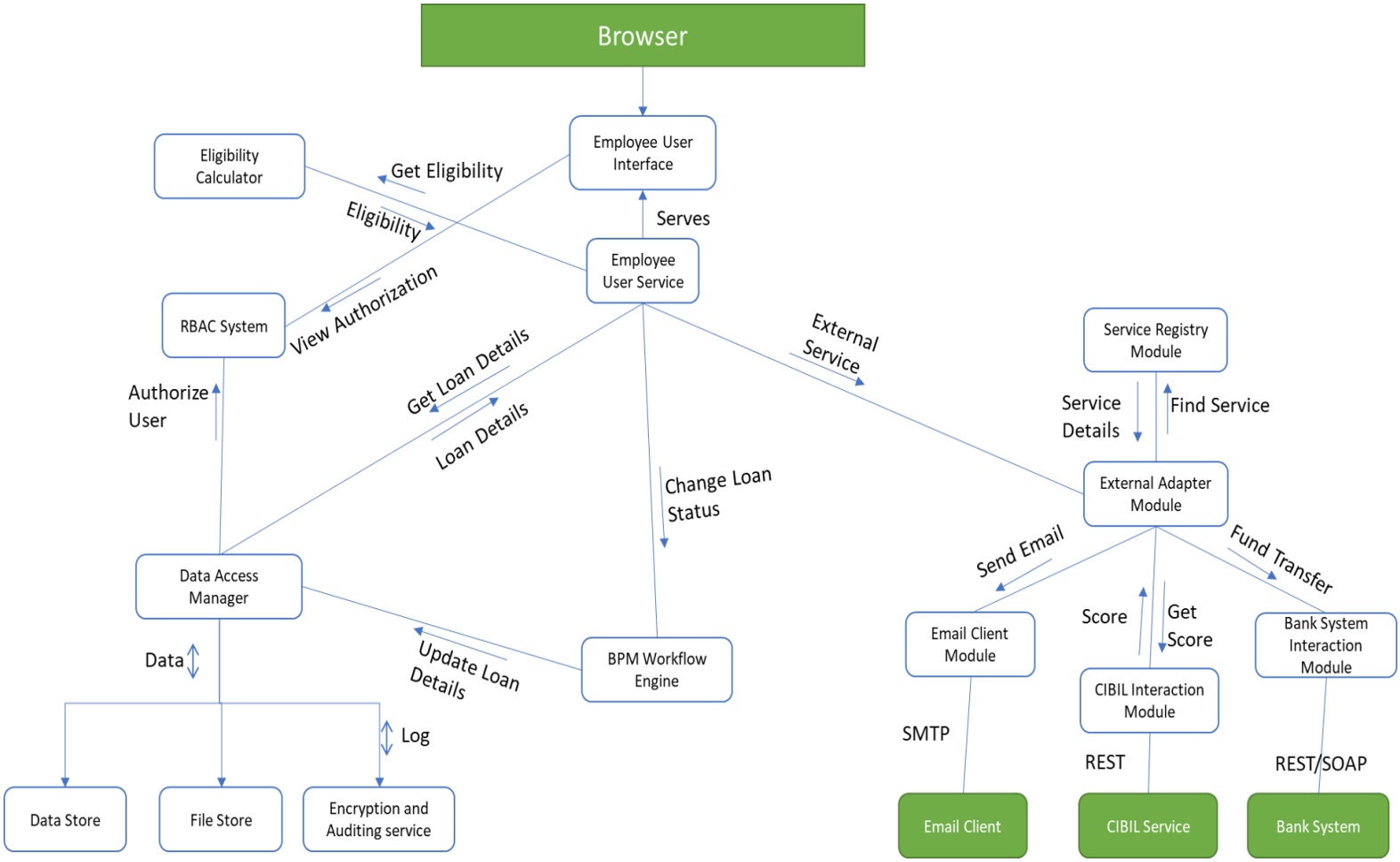
To avoid cluttering, I have broken the C&C diagram into two parts.

C&C Diagram Part 1



Bits ID: 2019ht13096
Name: Braj Kishor
Course No: S2-19_SSZG653
Course Title: Software Architectures

C&C Diagram Part 2



Bits ID: 2019ht13096
Name: Braj Kishor
Course No: S2-19_SSZG653
Course Title: Software Architectures

Q.7 Solution

The working of the system is described as follows:

1. Customers access the portal using browser and requests for loan application.
2. Customer is asked some basic questions like monthly income, current EMI, type of loan needed and amount etc.
3. The **customer service** with the above data, uses the **eligibility calculator** module and **Application type and form generator** module to generate appropriate form for the customer which is served in the UI.
4. The customer fills the application and uploads the documents needed. The **File Upload and Validation** module validates the document type and stores it in the data store using the **Data Access Manager**.
5. The **Data Access Manager** uses **RBAC service** and **Encryption and auditing service** to ensure that the data is accessed by verified **services/users** and all data is **encrypted** along with **Audit trails** maintained.
6. The **Employee User Interface Controller** uses **RBAC Service** to serve the correct and appropriate view to the Loan Officer and Finance Officer.
7. The **Employee User Service** uses **Data Access Manager** to show all the details of existing applications in the system. The Employee User Interface allows filtering, sorting and querying of loan application data.
8. Using the **BPM Workflow Engine**, an employee can change the status of the loan to various stages like ***more details required***, ***additional documents required***, ***CIBIL score requested***, ***Pending Approval***, ***Loan Approved***, ***Rejected, Disbursed, Pending Disbursal*** etc. The engine interacts with **Data Access Manager** to store the details into the data store.
9. For interacting with external systems, the **External Adapter module** is used. This adapter is an **interface** which has various implementation depending upon the external services configured. The Interface uses the **Service Registry** to discover various services configured in the system.

10. The **Email Client Module** uses Email Service to send and receive mails.
11. The **CIBIL score module** uses REST API to query the CIBIL score from the External CIBIL score services.
12. The **Bank System Interaction Module** is capable of interacting in both REST and SOAP to send request for disbursal of loan amount.

THE END