Department of Computer Science & Engineering
SOUTHEAST UNIVERSITY

CSE4056: Internship

Java EE-Web Development

A dissertation submitted to the Southeast University in partial fulfillment of the requirements for the degree of B. Sc. in Computer Science & Engineering

Submitted by

[Imran Khan]
ID:2014100000017
Batch: 37

Supervised by

Rezwan-Al-Islam Khan
Assistant Professor, Department of CSE
Southeast University

May, 2018

Letter of Transmittal

May 9, 2018

The Chairman,
Department of Computer Science & Engineering,
Southeast University,
Banani, Dhaka.

Through: Supervisor,Rezwan-Al-Islam Khan

Subject: Submission of Intenship Report

Dear Sir,

It is a great satisfaction to submit of my intenship report on the Web Development in Workspace Software Shop Limited.,93B,New Eskaton Road, Dhaka-1000,Bangladesh.It was a graeat pleasure to work as an intern engineer with such an exciting department of a renowned company.This internship report attempts to describe my observations and learning during the study of this course. I have tried my best to make this internship report widespread and informative as much as possible within the time allowed for me.I request your kind approval for this internship report to complete my degree.

Thank you.

Sincerely yours,

Md.Imran Khan
ID:2014100000017
Batch: 37rd Program: CSE

Supervisor:

Rezwan Al-Islam Khan
Assistant Professor, Department of CSE
Southeast University

Certificate

This is to certify that the intenship report titled “web Development” has been submitted by MD.Imran Khan in fulfillment of the requirments for the award of degree(B.Sc in CSE) in department of Computer Science and Engineering Southeast University.This report was carried out under my supervision.

Supervisor:

Rezwan-Al-Islam Khan
Assistant Professor
Department of CSE
Southeast University

Co-Supervisor:

Iftekhar Ishaque
Director of Web Developer
Software Shop Limited

Executive Summary

I had the opportunity ro do an interenship Web Development and Programming at Software Shop Limited.,93B,New Eskaton Road, Dhaka-1000,Bangladesh.Software shop Limited is a better position of company in Bangladesh.I have successfully completed my internship and submitted by insternship exprience and responsibilities during my internship period. Web Development and Programming is business management software that allows an organization to use a system of integrated applications to manage the business Web Development and Programming integrates all facets of action, including product planning,development,manufacturing processes, sales and marketing, security system.

Acknowledgements

I would like to thanks my supervisor Rezwan -Al-Islam Khan,Assistant Professor,Department of CSE for giving me his valueable guidelines support and encouragement.Also I would like to Sheikh Shamir Shakir, Software Specialist Enginner, in workspace Software shop Limited for helping me with valueable advice co-ordination for internship and in the preparation of this report.Thanks to all my friends and people who helped to prepare and complete this report.

Contents

List of Tables

List of Figures

0.1 Find BillInformation

0.2 Payment and Update BillInformation

0.3 Unpaid BillInformation

0.4 Cancel BillInformation

0.5 Current Date BillInformation

0.6 Acknowledgement API

0.7 Call Back API

0.8 Reconcilation API

....

Chapter 1

Introduction

1.1 Presentation of the company/organization

a. Brief history

Software Shop Limited (SSL Wireless) started its journey in 1999 as an associated company of one of the biggest garments exporting groups in Bangladesh, Concorde Garments. Since then it has come a long way contributing in the IT education sector, later from the year 2007, it revamped as SSL Wireless and started focusing on the domestic software development and telecommunication VAS business.

Over the last ten years, SSL has been focusing on the development of media and financial services and is at present serving 20+ TV and Online News Channels, with 45+ leading private and public banks, 12+ insurance companies, and 800+ corporate businesses of Bangladesh offering application software solutions including Mobile VAS, Mobile Financial Services, Messaging Solutions, Mobile Recharge, Online Payments, Internet Banking VAS and Payments Solutions, Agent Banking Services etc. We are also serving the government by providing SMS service to the Passport Office, BRTA and DESCO.

With its consistent efforts, SSL has been able to become the LARGEST Merchant Solution Provider serving 1000+ merchants across the country through its Internet Payment Gateway Solution called SSLCOMMERZ.

Out of the total online transaction in Bangladesh, 80% of those are processed through this gateway.

SSLCOMMERZ platform is also licensed and authorized as PSO (Payment Systems Operator) by Bangladesh Bank to operate its Payment Gateway services.

b. Nature of the organization/competition

Since 2009, SSL has been working as the authorized distributor of Symantec Inc, USA, and have been delivering Website Security Solutions to numerous banks, telecom, corporate and e-commerce sites.

SSL Wireless has been working as the local partner of CA Technologies, USA, for more than one year and offering 2FA solutions to the Payment Card Industry of the country. SSL has already signed agreement with 10 prominent banks which included Standard Chartered Bank, BRAC Bank, etc.

In the rise of latest cyber security threats in Bangladesh, SSL Wireless has partnered with AlienVault to bring an unique approach which can help the local organizations, banks and other enterprises to accelerate and simplify threat detection, incident response, and regulatory compliance. AlienVault provides worldś most renowned and reputed Unified Security Management (USM) solution.

We have been the pioneer in retailing TV contents showcasing over mobile WAP portals. SSL Wireless has been the first solution provider to offer Airtime Recharge & Bill Payment Service using Internet Banking Service. It was also the first to launch the country's first web based mobile top-up retailing service named www.easy.com.bd that was accepting local VISA, MasterCard and Dutch Bangla Bank cards as payment method. SSL is currently connected with all the Telecom Operators as the largest virtual distributor in Bangladesh.

c. Product lines

SSL Wireless has also launched its brand new platform called etunes for addressing the copyright issues of music industry.

Through etunes platform, artists can earn his/her deserved revenue by selling of their music through different channels like website, mobile site, app, caller tunes, etc.

The people at SSL are all well aware about their specific duty towards the society and the industry. This advantage along with the precise understanding of the local market lets SSL to serve its customers with the appropriate solutions and services. The following reasons have played a vital role in the rapid growth and expansion of SSL Wireless.

- Reliability of Service and customer confidence.
- High standard of professionalism of the employees.
- Maintenance of confidentiality and security of customers' data, especially for the banks, insurance and TV channels.
- With consistent marketing efforts SSL has created brand equity value.
- Capability of working in both Online and Mobile sector.
- Technical capability to integrate the solutions and to make it accessible to the target customers.

For its outstanding efforts in innovation of new mobile services for Telecom Operators and TV channels, SSL Wireless has received the "Best Innovative Service 2010" Award from GrameenPhone. Its contribution was also recognized by "mBillionth Award South Asia 2011" in the areas of Mobile Banking Services, which was held in New Delhi, India. Its e-commerce service brand "SSLCOMMERZ" was also nominated for excellence in online payment service in the "National Digital Innovation Award South Asia 2011" that was held in Bangladesh.

etunes, SSL's platform for music distribution and production has won the BRAC-Manthan Award 2016 which was held in Bangladesh, followed up by its win in the prestigious South Asian event Manthan Award 2016.

d. Business volume/ employee

The number of employee of Software Shop Limited is 250+.The employee of Software Shop Limited has Several part.Such as Web developer , Mobile apps developer , Service , Accounts , Architecture , Embeded. The values are Trust, Quality and Commitment.

1.2 Objective of the Internship

The purpose of this study is comprehend the various service and deeds in Workspace Software Shop Limited. Also understand the job activities as an engineer in Workspace Software Shop Limited.Submit recommendations on the substratum of findings. Accumulate pratical expreience base on the theoretical erudition. Observe the main role of software companies.Another important objective the relationship between client and company.Overall lerning and acquainted the academic purpose.

The main objective of the study is to evaluate the online and manual electricity bill payment system. The study was also conducted to assess the monetary and nonmonetary benefits of the new online bill payment process provided by DESCO.

Specific Objectives of the study are:

- To measure the amount of time and cost saved on average of beneficiaries of DESCO online bill payment system.
- To determine the number of visits saved on average of beneficiaries of DESCO online bill payment system.
- To assess the beneficiaries satisfaction; their opinion regarding the service; and other benefits of DESCO online bill payment system.

1.3 Methodology of study

The assignment adopted several methods to collect quantitative and qualitative information in order to fulfill the objectives of the study. Data were collected through phone interview, face-to-face interview, and in-depth interview from different groups of respondents.

This study was carried out to empirically investigate the factors affecting the consumerś intention to adopt an online utility bill payment system.
However, the survey sample of the study consisted only of physicians working in four (4) private hospitals situated within the Colombo city limits.

Theoretical sessions alone cannot make a student efficient and perfect in handling the authentic life professional situation.Methodology is important for us to comprehend the application of theoretical Knowloedge in practical fields through internship program.This program refers two parts organization part and project part.In organization part,to introduce the internee with all structure , functions,and performance of the organization.And in project part.pertaining to a particular quandary or a probing topic matching with capacity, interest of the organizations requisites.

Chapter 2

Literature Review

2.1 Enterprise Solution

2.1.1 PHP:Zend

Zend Framework is a collection of professional PHP packages with more than 244 million installations. It can be used to develop web applications and services using PHP 5.6+, and provides 100% object-oriented code using a broad spectrum of language features.

Zend Framework uses Composer as a package dependency manager; PHPUnit to test all packages; and Travis CI as a Continuous Integration service. Zend Framework also follows PHP-FIG standards, and includes an implementation of PSR-7 for HTTP message interfaces (as shepherded by Matthew Weier O'Phinney, Zend Framework project lead).

Zend Framework 3 evolved from both Zend Framework 2 and 1; cumulatively, the previous versions were downloaded more than 15 million times.

The principal sponsor of Zend Framework is Zend, a Rogue Wave Company, but many others have contributed components or significant features to the framework. Companies such as Google, Microsoft, and StrikeIron have partnered with Zend to provide interfaces to web services and other technologies they wish to make available to ZF developers.

Zend Framework 3 could not deliver and support all of these features without the help of the vibrant community. Community members, including contributors, make themselves available on mailing lists, IRC channels and other forums. Whatever question you have about Zend Framework, the community is always available to address it.

Zend Framework is an open source, object-oriented web application framework implemented in PHP 7 and licensed under the New BSD License. The framework is basically a collection of professional PHP based packages

Zend is the one and only framework which is best suited for Enterprise Projects. No other framework can match upto its flexibility, scalability and robustness. However, this framework is not meant for Beginners and even Intermediates but it can be of benefit to only the niche which is absolutely expert in PHP Programming as well as Object Oriented Concepts.

Zend Framework is an open source framework for developing web applications and services with PHP 5. Zend Framework is implemented using 100% object-oriented code. The component structure of Zend Framework is somewhat unique; each component is designed with few dependencies on other components.

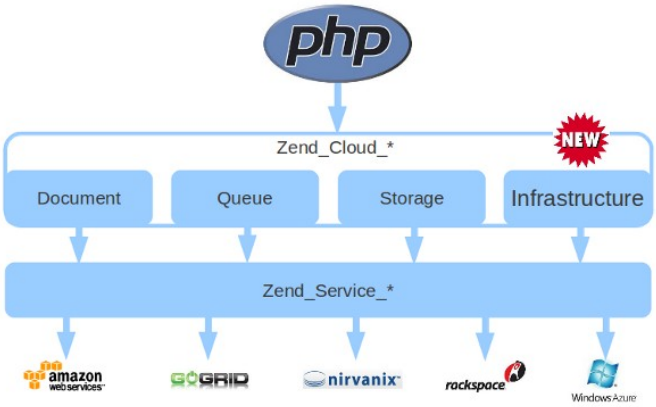

Figure 2.1: Zend stature

Exclusive Features:

- Offers the most efficiently designed workflows and API-centric applications that deploy amazingly fast.
- Agile Design methodology that focuses on delivering the high quality enterprise apps.
- World class Cloud support on servers. No Cloud lock-in, get everything on your choice of clouds.
- Inherently excellent streamlining and automation process that brings software from codes to production.
- The most coolest Drag and Drop editor, and unmatched front-end support (HTML, CSS and JavaScript) at your ease.
- Instant Online Debugging and PHP Unit testing tools to assist with codes and test case report.
- Intuitively connected Database Wizard and Editable Table Viewer for exclusive database connectivity.

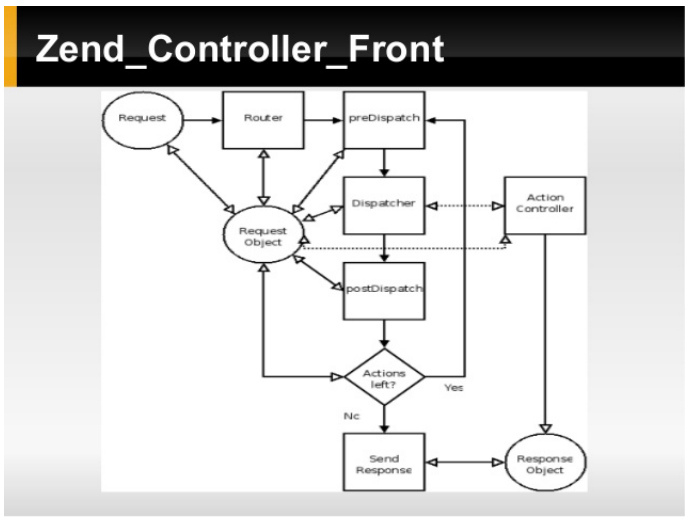

Figure 2.2: Zend stature

2.1.2 .NET

The .NET Framework is a class of reusable libraries (collection of classes) given by Microsoft to be used in other .Net applications and to develop, build and deploy many types of applications on the Windows platform including the following:

- Console Applications
- Windows Forms Applications
- Windows Presentation Foundation (WPF) Applications
- Web Applications
- Web Services
- Windows Services
- Services-oriented applications using Windows Communications Foundation (WCF)
- Workflow-enabled applications using Windows Workflow Foundation(WF)

That primarily runs on the Microsoft Windows operating system

.NET Framework is a software framework developed by Microsoft that runs primarily on Microsoft Windows. It includes a large class library named Framework Class Library and provides language interoperability across several programming languages.

Visual Overview of .NET Framework

Compiling a .NET program

What really happens when we compile a .NET program?

- The exe file that is created doesn't contain executable code, rather it's MicroSoft Intermediate Language (MSIL) code.
- When you run the EXE, a special runtime environment (the Common Language Runtime or CLR) is launched and the IL instructions are executed by the CLR to the machine language.
- The CLR comes up with a Just In Time Compiler that translates the IL to native language the first it is encountered.

So the process of programming goes through like:

- 1.We write a program in C, VB.Net and other languages.
- 2.We compile our code to IL code based on the language compiler (csc.exe, vbc.exe and so on).
- 3.Run your IL program that launches the CLR to execute your IL, using its its JIT to translate your program into native code as it executes.

Framework Component :

- Microsoft.NET Framework Runtime (.NET)

The Microsoft.NET Framework Runtime (.NET) is used for the development and production of .NET applications, in other words development means where the application is being made whereas production means the application is being deployed for production.

- Windows SDK

  The Windows SDK is used in applications for Windows XP, Windows Vista, Windows 7, Windows 8, Windows Server 2008, .NET Framework 3.0, .NET Framework 3.5 and .NET Framework 4.0.

- Visual Studio.NET

- It's an Integrated Development Environment and Rapid Application Development (drag and drop rather than using a text editor) to develop computer programs for Microsoft Windows, as well as web sites, web applications and web services. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed Code.

- Not needed in a production environment.

- Can be used in any type of .NET application.

Core of .NET Framework:

I will explain these topics in depth in my next articles. Just for a reference these are the core of the .NET Framework:

- Application Services.
- .NET Framework Base class Libraries
- Common Language Runtime.
- Garbage Collector.
- Common Type System.
- Common Language Specification.

The .NET Framework is a technology that supports building and running the next generation of apps and XML Web services. The .NET Framework is designed to fulfill the following objectives:

- To provide a consistent object-oriented programming environment whether object code is stored and executed locally, executed locally but Internet-distributed, or executed remotely.
- To provide a code-execution environment that minimizes software deployment and versioning conflicts.
- To provide a code-execution environment that promotes safe execution of code, including code created by an unknown or semi-trusted third party.

- To provide a code-execution environment that eliminates the performance problems of scripted or interpreted environments.
- To make the developer experience consistent across widely varying types of apps, such as Windows-based apps and Web-based apps.
- To build all communication on industry standards to ensure that code based on the .NET Framework integrates with any other code.

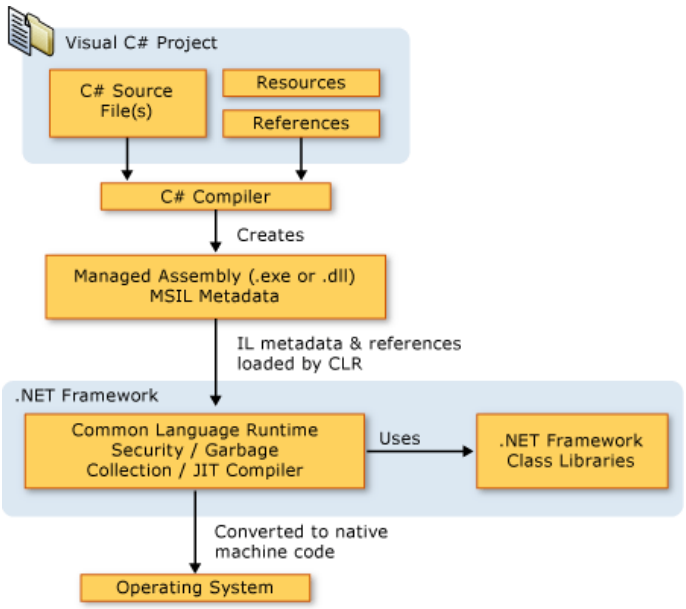

Figure 2.3: .NET stature

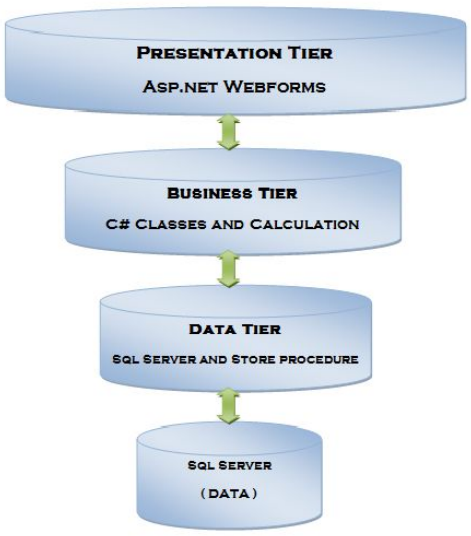

Figure 2.4: .NET Architecture

2.1.3 Java EE

There is no doubt that Java is one of the programming languages that is most in demand these days. Here, we have scooped a handful of good info about trending frameworks that ease everyday lives of software developers around the globe.

As per a survey conducted by RebelLabs, the media ally for the online Java User Group (Virtual JUG), the lineup of best java frameworks for 2018 is the following:

We will not push you or make you jump to conclusions just based on the comparison chart below, but we'll present you with the overview of each concept, letting you analyze and contrast the differences and pick the perfect match depending on your project.

1. Spring MVC:

Old but gold, Spring MVC is still ahead of the curve after more than a decade since its first release. After its expansion to embrace complete MVC framework, Spring kept on evolving adopting changes and turned into a full-scale Java framework for Internet-facing applications, offering software engineers a powerful toolkit for web application development and application configuration as well as for security projects. Being one of the most favored among programmers, Spring has a well-developed ecosystem, offering a variety of add-ons to choose from: REST APIs or SOAP web services, security (authentication and authorization, encryption), etc. Another benefit of this choice is that a big company of fellow developers who chose to use this particular Java framework, too. Thus, they are ready to give you a helping hand with any query you might have.

Pros and Cons:

Spring is not nesting in the top of the charts without reason. It is one of the best and most well-known Java frameworks because of:

- Simplified injection of test data through the use of POJOs.
- Enhanced modularity, resulting in better code readability.
- Loose coupling between different modules.
- Dependency Injection (DI) flexible use.

Although not a direct framework downside, regardless of the piles of reference documentation, Spring MVC might be more than newbie Java devs can chew, making the learning curve too steep. Thus, dealing with some parts of this Java framework can be tough for if you bump into a poorly documented chunk of references. The info you find might be different across versions.
By the way, due to our engineers' great experience, Spring MVC is one of our most fancied frameworks.

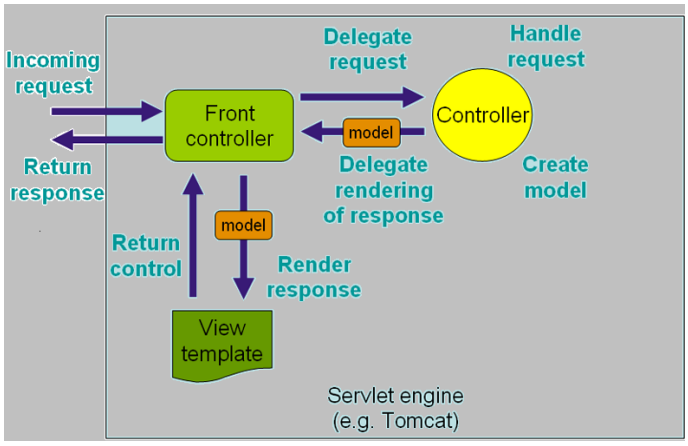

Figure 2.5: Spring

2. Struts2:

To elaborate even more on existing Java frameworks that are widely used by modern software engineers, we decided to refer to the successor of Apache Struts1, Struts2. This Java framework is quite a find for engineers who work with building contemporary Java EE web apps. The Apache Software Foundation has equipped developers with an extensive tool for creating enterprise-ready, web-facing applications, optimizing the development process from the beginning to the final point and post-go-live maintenance. This concept is the right choice if you sweat over high-load systems such as broadcasting portals.

Testing the written code takes less effort because of Struts2 implemented Action class plain POJO objects. Coupling becomes easier with the frameworks API, helping to tune Interceptors.

Pros and Cons:

When you ask yourself why you should consider Struts2, consider the luxury of not having to override or implement the Action class it is already in place! Repetitive setup code that IT engineers need to create for several actions is no longer a concern because Interceptors take care of it.

However, you might get turned off by the weight of this Java framework: some programmers find its not agile enough to go for.

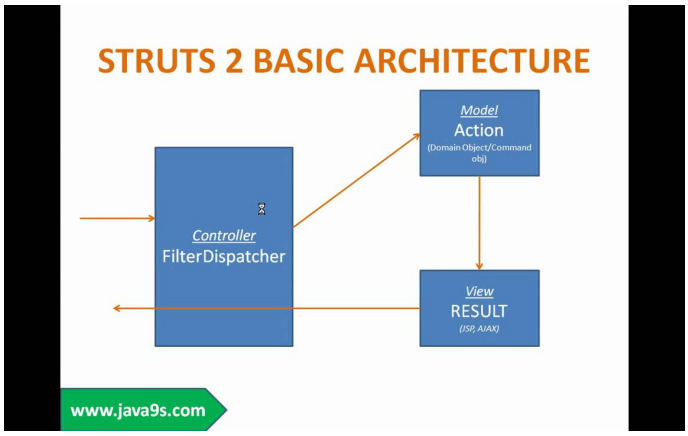

Figure 2.6: Strusts2

3. Hibernatehibernate-training:

Although not on RebelLabs list either, it is worth mentioning Hibernate when debating the best Java framework. This mapping Java framework cracks object-relational impedance mismatch issues by substituting persisting DB accesses high-level object handling functions. With every enterprise application being one of a kind, Hibernate is equipped with a great feature suite that helps backend developers fine-tune data access layer. This is an advanced ORM framework that lets you perform the database operation on Java objects (entities). It is a persistence concept that utilizes Hibernate cache to persist the data from the Java environment to the database.

Pros and Cons:

Hibernate allows you to communicate with any database using tiny alterations in the code you write, bridging the gap between objects and relational words. On top of that:

- Oracle, Db2, or MySQL, Hibernate is DB independent.
- Caching mechanism to bug database with similar queries.

- N+1 or Lazy loading support.

Despite its obvious advantages, Hibernate does not permit multiple inserts, not allowing some queries that JDBC does. In addition, this Java framework might not be a good idea for small-scale projects that have fewer tables.

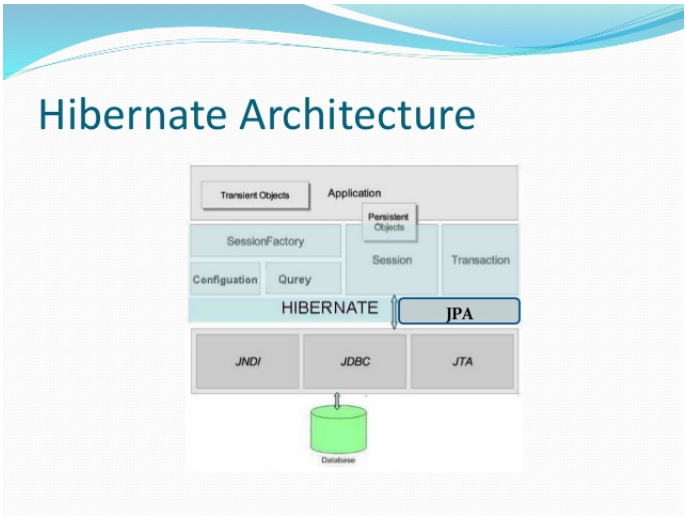

Figure 2.7: Hibernet Stature

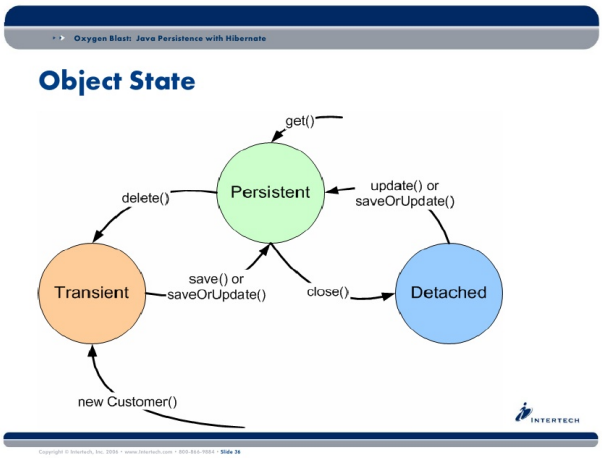

Figure 2.8: Hibernet Repository

4. JSF:

Being a part of Java EE, JavaServer Faces is supported by Oracle. Although this one is not the best frameworks for speedy Java development, it is easy to utilize because of great documentation provided by Oracle. Additionally, until you leave the Java EE environment, JSF has no external dependencies, but plenty of capabilities instead. Rich libraries and tools (including a toolset for easy user interface creation) become a magic wand that can back you up no matter how complex the application might seem.

Pros and Cons:

As mentioned, JSF is an integral part of Java EE. Thus, developers draw on an IDE software suite as well as a prime level of support from an industry-leading vendor.

On the contrary, JavaServer Faces is too vast to easily comprehend for those whose prior experience and skills in web development basics are not solid enough.

5. Google Web ToolkitGwt:

GWT is an another free Java framework allowing coders to create and optimize sophisticated web-based apps. The GWT Software Development Kit features core Java APIs and widgets for constructing apps compiled to JavaScript after.

Pros and Cons:

With Google Web Toolkit, you get single code base (both browser- and server-side programming in Java) that eases up the development and debugging processes. Simple integration with other technologies allows programmers embed GWT widgets in web pages. Custom widget capability is another benefit that can be created with the help of GWT.

However, if you are a web designer, GWT might be not the best Java framework to opt for; it is not amiable to those favoring plain HTML and late dynamic content placement.

2.1.4 Why Chose Spring Framework

1. Use Spring Framework for writing web applications:

It did not start out this way. But Spring MVC is now the de facto framework for java web applications development.

Most of my projects used Spring MVC. Other frameworks in space include Struts, JSF, Tapestry, GWT, Play etc.

Most other web applications provide integration with Spring Framework. With Spring MVC, this support is natural and built-in.

And Spring MVC delivers....

You are not bound to any container lifecycle that you need to manipulate. Spring does this behind the scenes for you. If you do decide to participate in the lifecycle, Spring exposes methods that can be overridden.

Spring MVC provides special form tags and validation out of the box. It supports the easy availability of model. It handles the request to model mapping.

You can add custom converters to handle special data in your forms. I will discuss this in a new light with code and samples soon.

Spring MVC makes sense for most of the web based application's. If you have a stateful application, I suggest you look at Spring Web Flow.

Only one of my projects leveraged Spring Web Flow. That was a long time ago.

Spring Web Flow lets you build a wizard a set of screens a user must go through before saving the data. Spring Web Flow will save you from having to store a lot of temporary data.
A mortgage application would be a stateful application. A flight reservation system is another example.

2. Use Spring Framework for exposing RESTful services:

Use Spring MVC to expose your services as RESTful URLs. Jersey is another alternative.

Spring HATEOAS sits on top of Spring MVC and helps it better represent resources.

HATEOAS stands for Hypermedia as the Engine of Application State.

I will try and give a quick and dirty explanation for HATEOAS. I understand the concept, but I have not used it in my projects yet.

A web services developer provides a contract to its consumer before they can call it.

In SOAP world, this was the stub or the interface that let the client know which methods were available.

In REST, we have been building URLs with appropriate data to send. We still needed to know what the server expects here.

In Hypermedia, the resource identifiers will give you a clue about what it

expects. This will give you insight into how you can further interact with it. A small convention over configuration on the web, if you will.

I used Jersey for one of my projects as opposed to Spring MVC. Jersey follows JSR specification for creating RESTful web services.

Spring MVC deviates from the specification in a slight way. It is easy to convert the API from Spring MVC to Jersey and vice versa. I will discuss MVC in future but for now it stands for Model, View and Controller. MVC helps define a clear separation of concerns amongst layers. For instance, your view should not depend on how java models it.

In my opinion, when in doubt always go with Spring MVC. If you ever need to support a web page with JSP, Jersey can not help you. Jersey deals with REST only. Spring MVC will give you best of both worlds.

If your client wants to use jQuery on top of JSPs, you can do so with Spring MVC. If, at a later point of time, you want to use a Javascript MVC framework, Spring MVC will continue to support you.

Few popular Javascript MVC frameworks are AngularJS, Ember, and Backbone.js

Javascript MVC frameworks bring structure to the javascript code. It keeps the code modular and clear.

CSS Frameworks bring ease and structure to the web design code. Bootstrap is the most popular CSS Framework out there.

Foundation is another one.

Spring MVC brings structure to the servlet programming model. It keeps the data model separate from view. Servlets embedded HTML code a long time ago. Struts MVC Framework solved that problem. Spring MVC came later.

Javascript MVC frameworks are like Spring MVC but on the javascript layer. With Javascript MVC frameworks, you build your web pages separate from your server logic.

The two layers then communicate and exchange data with servers via RESTful URLs.

More on MVC frameworks in future posts.

3. Use Spring Framework to secure your web applications:

For all my projects that involved web development, I have used Spring Security. A two-minute introduction is below.

Spring Security intercepts web URLs. Here are few examples:

You can tell it that all URLs with a specific pattern e.g. /static are accessible to all users.

You can tell it that all URLs with a specific pattern e.g. admin are only available to users with a specific role. e.g. ROLE_ADMIN

You can also add annotations on the java methods to make them secure.

Spring Security provides both authentication as well as authorization.

Authentication helps identify who you are.

Authorization helps identify what you can do in the system.

When starting a fresh project, I integrate spring security and use in-memory authentication.

In-memory authentication premise is simple. You hard code a bunch of username, password, role combination into a file or within your XML or java code.

Spring Security would then let you in using either one of them and give you the role for that user. This helps developers continue with the application development.

Later, you could plug in a different authentication provider. And there are no code changes to the business logic as a result of this.

4. Use Spring Framework for communicating with databases:

For most of my projects, I used Spring ORM with hibernate underneath. One of the projects used Spring JPA while the others were direct hibernate calls. ORM is the Object Relational Mapping. It is a way to represent your tables into java objects and vice versa. Hibernate is the most popular ORM Framework. myBatis is another one.

And if you are dealing with a lot of existing plain old JDBC code, my suggestion would be to use Spring JDBC.

Let Spring JDBC close your statements and connections. Use Spring JDBC to integrate connection pooling. Let Spring handle your database transactions as well.

I also ran into situations where I was using hibernate for ORM but had to call a plSQL procedure in Oracle. Hibernate expects your procedure in a certain way. So, I used Spring JDBC to call procedures if they did not fit in the structure required by hibernate.

If you are starting a new project, I would recommend using Spring Data. It sits on top of Spring ORM.

With Spring Data, you do not write standard CRUD operations. They come built-in. I will explain this in a future post.

I have used Spring Data for Oracle and MongoDB now without any issues. I know it has support for other NoSQL and graph databases like Redis, Cassandra, Neo4j, Couchbase, Elasticsearch and Solr.

5. Use Spring Framework to handle external resources or systems you have to work with:
In one of my projects, Our client would send us reports as XML files via web services. These XMLs were small, but there was no definite way to tell when these files would come in. We had 100k XML files come in six months on average.

Our goal was to intercept these files as they came in and process them for our system intake.

This is a typical scenario for a service that depends on an external system.

We used Spring Integration for this task. We set up a file poller for the directory of incoming files. Once the file came in, the event would get triggered. We would then push it through our workflow.

The code to handle the incoming file would stay separate from the rest of the workflow. This kept the external party dependency away from the rest of the system and made testing easy.

6. Use Spring Framework for testing purposes:

I once worked on a Struts 1.x based web application with EJB3 implementation underneath. They needed to perform some clean up of the data for their testing environment on a regular basis.

So, I created a simple standalone Spring JDBC project and ran a bunch of updates on the tables. I learned more about the project and data structure via this effort than from few months code sifting.

7. Use Spring Framework for standalone java projects:

I worked on a product that was a standalone CLI (command line interface) sold to clients. It was Java based business rules engine with heavy XML parsing.

When things went wrong, it was hard to test which component failed to deliver.

So, I introduced Spring Framework in a test package within the project. I used Spring Context, Spring Test, and JUnit to call these different components.

The testers had an XML and unit tests that could prep the data before each call.

8. Use Spring Framework to convert your application into an executable:

Going back to the Spring Batch project. We were kicking off these jobs from a web page based on users input.

The application was using Spring Framework and its different modules already. But an external team needed to kick off these batch jobs from their end on their own schedule.

So, I created an executable using Spring Boot to point to the main Spring Batch Job. It was the same job that we were calling from our web application. It was the same codebase.

The external team could call this executable nightly or whenever now. There was no code change needed from our side.

9. Use Spring Framework to integrate Social Media into your applications:

Social leverages OAuth underneath to make this possible. These social media accounts expose OAuth APIs to allow authentication.

Spring Mobile allows you to customize your pages so you can tweak them for mobile users. It implies having to change the code to accommodate both users.

This is cumbersome and not the best bang for the buck.

If I were starting out, I would leverage Bootstrap and Angular JS to build out the application.

Bootstrap is responsive CSS Framework and Angular JS is mobile friendly javascript MVC Framework.

As I write this, Angular 2.x is in works. Angular2 is a mobile-first platform.

10. Use Spring Framework to come up with a quick prototype:

If you are looking to expose few database tables as a set of screens, I recommend you look into Spring Roo. You will get a working CRUD application that will maintain data integrity out of the box.

You can then use the app to allow data manipulation without going into the database.

Sometimes you want to ask your client for important columns to show on screen. Spring Roo could be a quick prototype for client feedback in that case. You can do this with Grails even faster. If you know Groovy, maybe you would consider this.

Today Spring includes all the followings:

Spring Core Container: This is the base module of Spring and provides spring containers (BeanFactory and ApplicationContext).

Aspect-oriented programming: enables implementing cross-cutting concerns.

Authentication and authorization: configurable security processes that support a range of standards, protocols, tools and practices via the Spring Security sub-project (formerly Acegi Security System for Spring).

Convention over configuration: a rapid application development solution for Spring-based enterprise applications is offered in the Spring Roo module.

2.2 Database

2.2.1 PostgreSQL:

PostgreSQL is an open-source, object-relational database management system (ORDBMS) that is not owned or controlled by one company or individual. Because postgresSQL software is open-source, it is managed mostly through a coordinated online effort by an active global community of developers, enthusiasts and other volunteers.

First released in the mid-1990s, postgresSQL is written in C. Its primary competitors include Oracle DB, SQL Server and MySQL.

This term is also known as Postgres.

PostgresSQL and ingres, an earlier effort, were both developed by a team at the University of California at Berkeley. PostgresSQL did not originally support structured query language (SQL) – QUEL query language was used until 1994, when SQL support was added. In 1996, the first official open-source software version of PostgresSQL was released.

PostgresSQL supports almost all relational database features and offers a few unusual features that are normally absent in other RDBMS engines. Commonly supported objects include views, stored procedures, indexes, triggers and object-defined data types, in addition to general RDBMS features such as primary keys, foreign key relationships and atomicity.

Certain critical postgresSQL features are similar to Oracle DB and other database engines; such features include the use of concepts like tablespaces, save-points and point-in-time recovery.

PostgreSQL features highlights PostgreSQL has many advanced features that other enterprise database management systems offer, such as:

- User-defined types
- Table inheritance
- Sophisticated locking mechanism
- Foreign key referential integrity
- Views, rules, subquery
- Nested transactions (savepoints)
- Multi-version concurrency control (MVCC)
- Asynchronous replication

The recent versions of PostgreSQL support the following features:

Native Microsoft Windows Server version

- Tablespaces
- Point-in-time recovery

2.2.2 MongoDB:

MongoDB is a cross platform and open source document-oriented database, a kind of NoSQL database. As a NoSQL database, MongoDB shuns the relational databases table based structure to adapt JSON documents that have dynamic schemas which it calls BSON.

This makes data integration for certain types of applications faster and easier. MongoDB is built for scalability, high availability and performance from a single server deployment to large and complex multisite infrastructures.

MongoDB was first developed by MongoDB Inc., known then as 10gen, in October 2007 originally as a major part in a PaaS (Platform as a Service) product similar to Windows Azure and Google App Engine. The development was shifted to open source in 2009.

MongoDB became one of the most popular NoSQL databases, being used as the backend for many major websites including eBay, Craigslist, SourceForge and The New York Times. MongoDB is available under the GNU Affero General Public License while its language drivers are available under the Apache License. There are also commercial licenses being offered.

MongoDB features:

- Ad hoc queries -
  supports search by field, regular expression searches, and range queries.

- Indexing -
  any field in the BSON document can be indexed.

- Replication -
  provides high availability via replica sets which consists of two or more copies of the original data.

- Load balancing -

  sharding is the method used to allow MongoDB to scale horizontally, meaning that data will be distributed and split into ranges and then stored in different shards which can be located in different servers. Shard keys are used to determine how the data will be distributed.

- Aggregation -

  MapReduce can be applied to enable batch processing of data as well as perform aggregation operations.

- File storage -

  MongoDB can be used as file system which makes use of the above functions and acting in a distributed manner through sharding.

2.2.3 MariaDB:

MariaDB is based on SQL and supports ACID-style data processing with guaranteed atomicity, consistency, isolation and durability for transactions. Among other features, the database also supports JSON APIs, parallel data replication and multiple storage engines, including InnoDB, MyRocks, Spider, Aria, TokuDB, Cassandra and MariaDB ColumnStore.

Much of the development work on the open source database has focused on achieving feature parity between MariaDB and MySQL. MariaDB Corp., the driving force behind MariaDB, says that "for all practical purposes" the database is binary-compatible with MySQL. As a result, most users can switch between the two technologies simply by uninstalling MySQL and installing MariaDB in its place.

Nonetheless, there are some incompatibilities between corresponding versions of the databases. For example, in version 10.1 and above, MariaDB stores JSON data in a different format than MySQL 5.7 does. To compensate, users looking to replicate columns of JSON objects from MySQL to MariaDB need to either convert them to the format used by the latter or run statement-based replication jobs via SQL.

MariaDB Corp., based in Espoo, Finland, with U.S. headquarters in Menlo Park, Calif., offers a commercial version of MariaDB on a subscription basis, along with complementary products and a set of training, remote management and migration services. However, the database's source code is maintained by the MariaDB Foundation, a group that was formed in 2012 to preserve the software's open source nature.

Origins and versions of MariaDB

Underlying the MariaDB effort is dissatisfaction on the part of MySQL's initial developers with the evolution of that database, which came under the stewardship of Oracle when the database market leader completed its purchase of Sun Microsystems Inc. in early 2010, nine months after announcing the deal.

The development of MariaDB was led by Michael "Monty" Widenius, who was also the lead developer on MySQL and a founder of MySQL AB, a vendor that Sun bought in 2008. After leaving Sun in early 2009, he and several colleagues began work on a MySQL storage engine that soon morphed into MariaDB, which is named after Widenius's youngest daughter. He also formed a new company that merged with a database consulting business to create what eventually became known as MariaDB Corp.

The first release of MariaDB, known as version 5.1, became available for production uses in early 2010. Versions 5.2, 5.3 and 5.5 followed, and then MariaDB 10.0 was released in 2014. It represented a change in the database's classification scheme, as earlier release numbers were patterned after MySQL ones.

MariaDB 10.1 and 10.2 came after that in 2015 and 2017, respectively. The 10.2 version, which was up to a 10.2.12 release as of January 2018, employs InnoDB as the default storage engine, and new features include a JSON data type designed to boost ties with MySQL on JSON. Next in line for release is MariaDB 10.3, which became available in alpha and beta versions in 2017.

2.2.4 SQL Server:

SQL Server is Microsoft's relational database management system (RDBMS). It is a full-featured database primarily designed to compete against competitors Oracle Database (DB) and MySQL.

Like all major RBDMS, SQL Server supports ANSI SQL, the standard SQL language. However, SQL Server also contains T-SQL, its own SQL implemention. SQL Server Management Studio (SSMS) (previously known as Enterprise Manager) is SQL Server's main interface tool, and it supports 32-bit and 64-bit environments.

SQL Server is sometimes referred to as MSSQL and Microsoft SQL Server.

Originally released in 1989 as version 1.0 by Microsoft, in conjunction with Sybase, SQL Server and its early versions were very similar to Sybase. However, the Microsoft-Sybase partnership dissolved in the early 1990s, and Microsoft retained the rights to the SQL Server trade name.

Since then, Microsoft has released 2000, 2005 and 2008 versions, which feature more advanced options and better security.

Examples of some features include: XML data type support, dynamic management views (DMVs), full-text search capability and database mirroring.

SQL Server is offered in several editions with different feature set and pricing options to meet a variety of user needs, including the following:

- Enterprise:

  Designed for large enterprises with complex data requirements, data warehousing and Web-enabled databases. Has all the features of SQL Server, and its license pricing is the most expensive.

- Standard:

  Targeted toward small and medium organizations. Also supports e-commerce and data warehousing.

- Workgroup:

  For small organizations. No size or user limits and may be used as the backend database for small Web servers or branch offices.

- Express:

  Free for distribution. Has the fewest number of features and limits database size and users. May be used as a replacement for an Access database.

2.2.5 MySQL:

MySQL is a fullfeatured relational database management system (RDBMS) that competes with the likes of Oracle DB and Microsofts SQL Server. MySQL is sponsored by the Swedish company MySQL AB, which is owned by Oracle Corp. However the MySQL source code is freely available because it was originally developed as freeware. MySQL is written in C and C++ and is compatible with all major operating systems.

MySQL was a free software database engine originally developed and first released in 1995. MySQL is named after My, the daughter Michael Widenius, of one of the products originators. It was originally produced under the GNU General Public License, in which source code is made freely available.

MySQL is very popular for Web-hosting applications because of its plethora of Web-optimized features like HTML data types, and because it is available for free. It is part of the Linux, Apache, MySQL, PHP (LAMP) architecture, a combination of platforms that is frequently used to deliver and support advanced Web applications. MySQL runs the back-end databases of some famous web-sites, including Wikipedia, Google and Facebook- a testament to its stability and robustness despite its decentralized, free-for-all philosophy.

MySQL was originally owned by Sun Microsystems; when the company was purchased by Oracle Corp. in 2010, MySQL was part of the package. Although MySQL is technically considered a competitor of Oracle DB, Oracle DB is mainly used by large enterprises, while MySQL is used by smaller, more Web-oriented databases. In addition, MySQL differs from Oracle's product because it's in the public domain.

MySQL is an open source relational database management system (RDBMS) based on Structured Query Language (SQL).

MySQL runs on virtually all platforms, including Linux, UNIX, and Windows. Although it can be used in a wide range of applications, MySQL is most often associated with web-based applications and online publishing and is an important

component of an open source enterprise stack called LAMP. LAMP is a Web development platform that uses Linux as the operating system, Apache as the Web server, MySQL as the relational database management system and PHP as the object-oriented scripting language. (Sometimes Perl or Python is used instead of PHP.)

MySQL, which was originally conceived by the Swedish company MySQL AB, was acquired by Sun Microsystems in 2008 and then by Oracle when it bought Sun in 2010. Developers can still use MySQL under the GNU General Public License (GPL), but enterprises must obtain a commercial license from Oracle.

Offshoots of MySQL are called forks. They include:

Drizzle a lightweight open source database management system in development based on MySQL 6.0.

MariaDB a popular community-developed "drop-in" replacement for MySQL that uses MySQL APIs and commands.

Percona Server with XtraDB an enhanced version of MySQL known for horizontal scalability.

5 REASONS TO CHOOSE MYSQL:

- Secure Money Transactions:

  MySQL transactions work as a single unit, which means unless and until every individual operational stage is successfully completed, the transaction is not cleared. So, if an operation fails at any stage, the entire transaction happening within that group fails. MySQL ensures that financial transactions have data integrity, so customers can make worry-free transactions online. The money is not debited until the entire process is completed and in case of failure, every process is reverted to the previous stage.

- On-Demand Scalability:

  MySQL comes with the advantage of unmatched flexibility that facilitates efficient management of deeply embedded applications, even in gigantic

data centers that stack tremendous amounts of mission-critical information. It enables complete customization to cater to the unique requirements of eCommerce businesses with a much smaller footprint. MySQL provides ultimate platform flexibility to enterprises who need additional features and functionalities for their database servers.

- High Availability:

  Consistent availability is the stalwart feature of MySQL enterprises that deploy it can enjoy round-the-clock uptime. MySQL comes with a wide variety of cluster servers and master-slave replication configurations that enable instant failover for uninterrupted access. Whether you run an eCommerce website or a high-speed processing system, MySQL is designed to process millions of queries and thousands of transactions while ensuring unique memory caches, full-text indexes and optimum speed.

- Rock-Solid Reliability:

  Protecting sensitive business information is the primary concern of every enterprise. MySQL ensures data security with exceptional data protection features. Powerful data encryption prevents unauthorized viewing of data and SSH and SSL supports ensure safer connections. It also features a powerful mechanism that restricts server access to authorized users and has the ability to block users even at the man-machine level. Finally, the data backup feature facilitates point-in-time recovery.

- Quick-Start Capability:

  You can go from software download to complete installation in just 15 minutes. MySQL is exceptionally quick, regardless of the underlying platform. It features self-management capabilities like auto restart, space expansion and automatic configuration changes for ease of management.

It also comes with a comprehensive set of migration tools and a fully loaded graphical management suite. MySQL enables real-time performance monitoring for timely troubleshooting of operational issues from a single workstation.

For all of these reasons, organizations are using MySQL to instantly develop and launch apps. From retail and finance, to healthcare and manufacturing, many industries are capitalizing on the cost-effectiveness, efficiency and reliability of MySQL to deliver seamless services and boost their revenue. But when it comes to optimizing MySQL deployments for performance and availability, enterprises face the following challenges, because scaling MySQL needs much more than just a database:

5 CHALLENGES OF SCALING MYSQL:

- Long Development Time:

  Scaling frameworks that cannot be optimized with master/slave setups requires extensive development time. Replication lag further complicates app logic because it disrupts the data consistency between the slave and the master. Finally, MySQL server modifications need constant coordination between database teams and apps.

- Replication:

  MySQL servers often run into replication conflicts during a manual failover when multi-master setups are involved.

- Database Logging Costs:

  Database logging is expensive and so it remains disabled most of the time. As a result, organizations lack real-time visibility into slow logs, which delays troubleshooting.

- Query Caches:

  MySQL server query cache is of little help when handling a high volume of workload, because cache invalidation cannot be controlled.

- High Connection Churn:

  If your apps rely on a LAMP stack, they tend to have a high volume of user sessions running concurrently and, consequently, they experience a high connection churn. So most of your valuable server resources are exhausted on connection management.

Some companies consider sharding as a scaling option, but sharding adds significant complexity, cost and its own set of additional challenges. The easiest way to leverage the powerful features of MySQL without making any modifications at the app level or writing a single line of code is to utilize database load balancing software.

The Solution? Database Load Balancing Software for MySQL

Database load balancing software solves the challenges that crop up when scaling your MySQL database. It enables automatic read/write split so you can leverage readable replicas with no code changes, augments failover to make it invisible at the user or app tier, provides unparalleled visibility into analytics, and facilitates one-click transparent caching. This intuitive, agentless approach increases response times up to 60X, prevents downtime during database failovers, doubles website performance and ensures better business outcomes. With database load balancing software, enterprises can upgrade their apps in an instant and ensure that they remain up and running round the year.

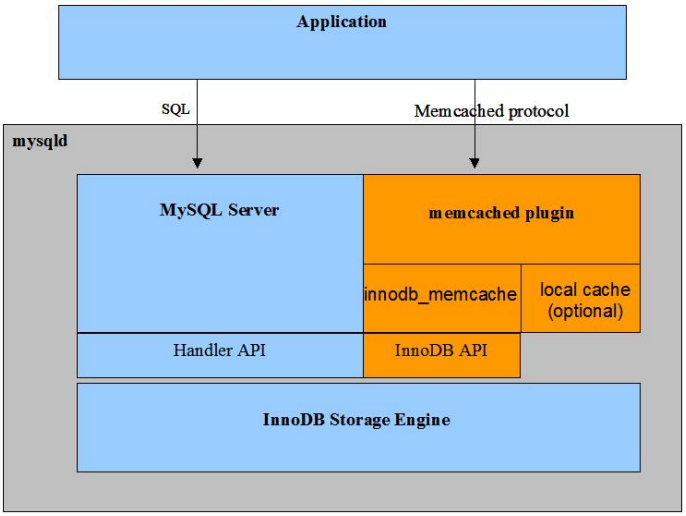

Figure 2.9: Mysql Stature

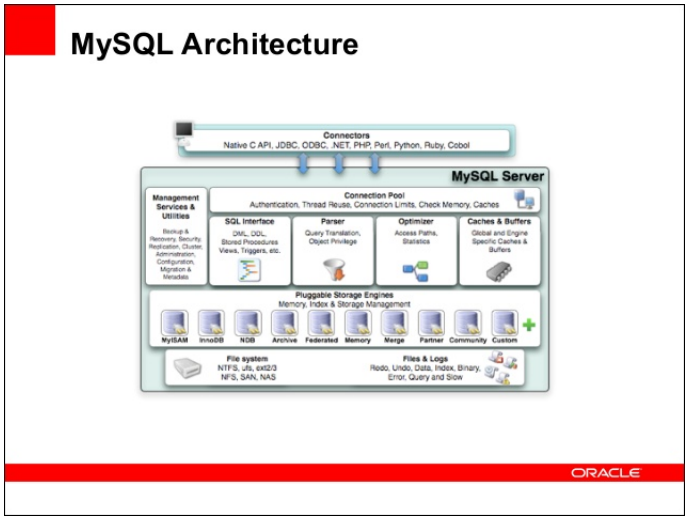

Figure 2.10: Mysql Architecture

Chapter 3

RESTful API

APIs are driven by a set of specific technologies, making them easily understood by a wide variety of developers.
A focus on simplicity means that APIs can work with any common programming language and be understood by any programmer, even one with little or no training in API technology.

RESTful:

Application Programming Interface (API) is a set of clearly defined methods of communication between various software components. A good API makes it easier to develop a computer program by providing all the building blocks. While the specifications vary between various APIs, the end goal is to provide value to the programmer through utilization of the services gained from using an API.

The most popular approach to delivering web APIs is Representational State Transfer (REST). This approach to API design takes advantage of the same internet mechanisms (based on the HTTP protocol) used to view regular web pages, so it has the advantage of faster implementations and is easier for developers to understand and put to use.

REST APIs allow you to take information and functionality that is already available on your website and then quickly make it available through a programmatic API, so that both web and mobile applications can reuse it, dramatically extending your company's reach over new channels, all without much additional work.

JSON:

JavaScript Object Notation (JSON) is a way for programs to exchange information. APIs are a way for programs to communicate, but since they don't have voices, they need a way to describe the data and information they that is being exchanged. JSON uses brackets, quotes, colons, and commas to separate data, giving the information meaningful structure so that computers can differentiate between a first name and last name or any other information that potentially describes data.

JSON has become one of the preferred ways for programmers to enable API communication. It provides a lightweight, simple way to exchange data across the internet while maintaining the structure and meaning of that data.

Security:

API security is the single biggest challenge organizations want to see solved in the years ahead, and solving the security challenge is expected to be a catalyst for growth in the API world.

According to research by SmartBear presented in their State of APIs Report 2016:

- Security is the 1 technology challenge teams want to see solved; 41.2 of respondents say security is the biggest API technology challenge they hope to see solved.

- Security is the 4 technology area expected to drive the most API growth in the next two years; 24 of API providers say digital security will drive the most API growth in the next two years.
- 40.4 of API providers are currently utilizing a tool for testing API security.

Keys:

This is not any kind of formal standard, but something like a common practice by API providers, and supported by API management providers - the usage of one or two keys that accompany every API call. API keys are really more about identifying the application and user rather than a true security protocol, but it is perceived as secure by many.

Public REST services without access control run the risk of being farmed, leading to excessive bills for bandwidth or compute cycles. API keys can be used to mitigate this risk. They are also often used by an organization to monetize APIs; instead of blocking high-frequency calls, clients are given access in accordance with a purchased access plan.

Typically, an API key gives full access to every operation an API can perform, including writing new data or deleting existing data. If you use the same API key in multiple apps, a broken app could destroy your users' data without an easy way to stop just that one app. Some apps let users generate new API keys, or even have multiple API keys with the option to revoke one that may have gone into the wrong hands. The ability to change an API key limits the security downsides.

JSON Web Token (JWT):

JSON Web Token (JWT) is an open standard, an extension of the OAuth 2.0, for creating access tokens that assert some number of claims.

Whereas API keys and OAuth tokens are always used to access APIs, JSON Web Tokens (JWT) can be used in many different scenarios. In fact, JWT can store any type of data, which is where it excels in combination with OAuth.

Webhooks:

Webhooks are a form of push notifications that can be triggered by a specific action. When triggered, they push information to an external website address. Webhooks allow developers to choose the action, URL, and fields associated with the webhook push. Once a webhook is triggered, it passes all associated information to the location where a developer can process it.

Webhooks became very popular, and, even in OpenAPI 3, you can define the format of the "subscription" operation as well as the format of callback messages and expected responses to these messages. This description will simplify communication between different servers and will help you standardize the use of webhooks in your API. Today, many tools help you create webhooks based on the OpenAPI spec and even helps you build your API definitions WYSIWYG.

Webhooks are a great way to reduce constant polling on an API because they push data to API developers only when the required action is triggered. Webhooks make the API a two-way street, allowing developers to not only pull data from API platforms but also receive data in real time as events occur.

List of Services:

The REST API of JasperReports Server responds to HTTP requests from client applications, in particular the following methods (sometimes called verbs):

- GET to list search and acquire information about server resources.
- POST to create new resources and execute reports.
- PUT to modify existing resources.
- DELETE to remove resources.

As with any RESTful service, not all methods (GET, PUT, POST, and DELETE) are supported on every service. The URLs usually include a path to the resource being acted upon, as well as any parameters that are accepted by the method. For example, to search for input control resources in the repository, your application would send the following HTTP request:

A "REST API" is two things: it's a web service and it's RESTful. By virtue of being a web service, you get some loose coupling. The client need not be aware of internal implementation details and there is, as you note, a real opportunity for platform/language independence.

Being RESTful offers additional benefits aimed at additional decoupling, so as to allow extreme scalability. REST forbids conversational state, which means we can scale very wide by adding additional server nodes behind a load balancer. The uniform interface means that we don't have to document on a per-resource or per-server basis, the basic operations of the API. The universal identifiers embodied by URIs mean again that there is no resource or server specific customs needed to refer to our resources, which assures that any tool that can work with HTTP (of which there are a LOT) can play ball with our service.

In particular, a client doesn't need to learn some custom naming convention. This makes uniform caching possible, using either expiry or validation (etags), and it gives us an optimistic locking paradigm through the use use of conditional requests.

The hypermedia constraint assures us also that application processing transitions are always navigable by clients simply by following opaque server provided links. When done correctly, this means that the client doesn't need to understand anything more than the data format. In particular, it doesn't need to understand the URI structure of the server. This is the hardest one to get correct, but it's also the one with the largest benefit. Ideally, a REST API would be processed by a set of client side language bindings, one for each of the data formats (media types). When there is a potential for widespread reuse of data formats (eg: Atom, SVG, html, css, etc...) the benefits become enormous as we aren't creating per-service solutions anymore. If you want to scale to internet scale with massive numbers of clients and services, it's just not reasonable to expect clients to do a lot of design on a per-server basis.

Native libraries, by comparison, are going to be much more tightly coupled. Often the protocols in question may allow stateful or domain specific communications that offer some performance improvement at the cost of complexity. Typically, if I implement a RESTful service, the internals of that service might be using a native library to do the work, but this is encapsulated from the service clients so that they see a scalable, low coupling, low custom-design solution to integrate with.

RESTful - REST - representational state transfer - considered a performance efficient alternative to the SOAP web services. REST is an architectural style, not a protocol.

- Performance : better performance efficiency compared to SOAP as they dont have strict spects, consuming less network bandwidth and resource allocation for information exchange
- Language and platform independence
- Support for versatile data formats : XML,JSON , etc

advantages of REST for development

- Separation between the client and the server: the REST protocol totally separates the user interface from the server and the data storage. This has some advantages when making developments. For example, it improves the portability of the interface to other types of platforms, it increases the scalability of the projects, and allows the different components of the developments to be evolved independently.

- Visibility, reliability and scalability. The separation between client and server has one evident advantage, and that is that each development team can scale the product without too much problem. They can migrate to other servers or make all kinds of changes in the database, provided the data from each request is sent correctly. The separation makes it easier to have the front and the back on different servers, and this makes the apps more flexible to work with.

- The REST API is always independent of the type of platform or languages: the REST API always adapts to the type of syntax or platforms being used, which gives considerable freedom when changing or testing new environments within the development. With a REST API you can have PHP, Java, Python or Node.js servers. The only thing is that it is indispensable that the responses to the requests should always take place in the language used for the information exchange, normally XML or JSON.

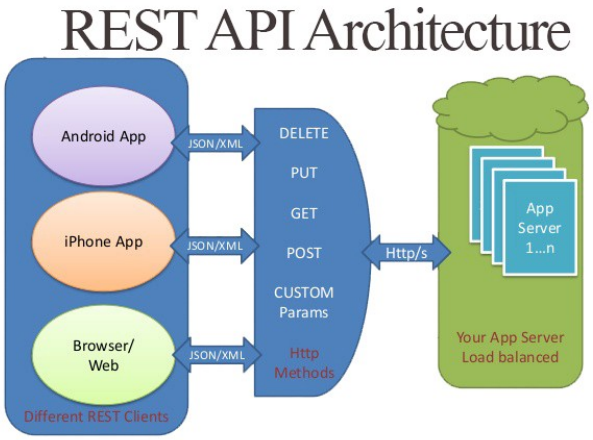

Figure 3.1: RESTful API Architecture

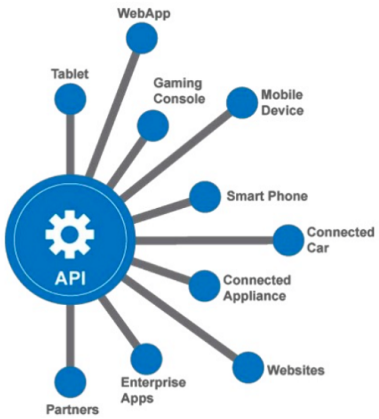

Figure 3.2: RESTful API Stature

Chapter 4

Details of Internship Work

I work RESTful api in my project for online Billing System . There are all api details i use in my project:

1. Get Bill Information

request parameters:

billNumber,userName,Password.

response data : data from billInfo table against given billNumber

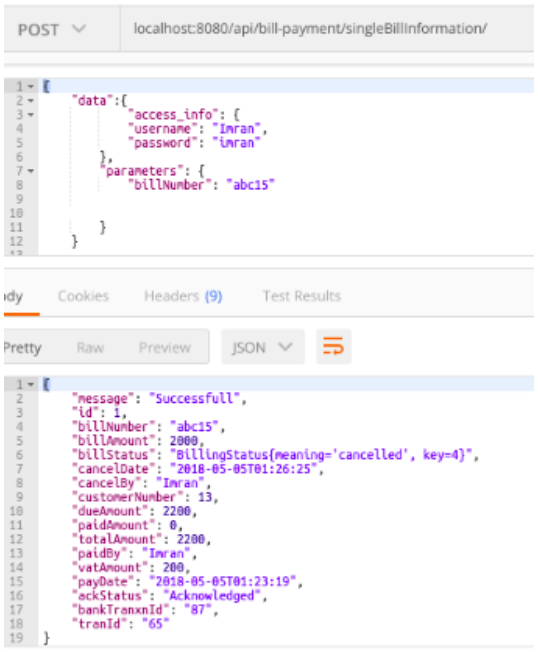

Figure 4.1: Find one Bill Information

2. Bill payment API

request parameters:

billNumber, amount,tranID,userName,password.

Update billInfo table accordingly

response data: status, message,banktranID,tranID.

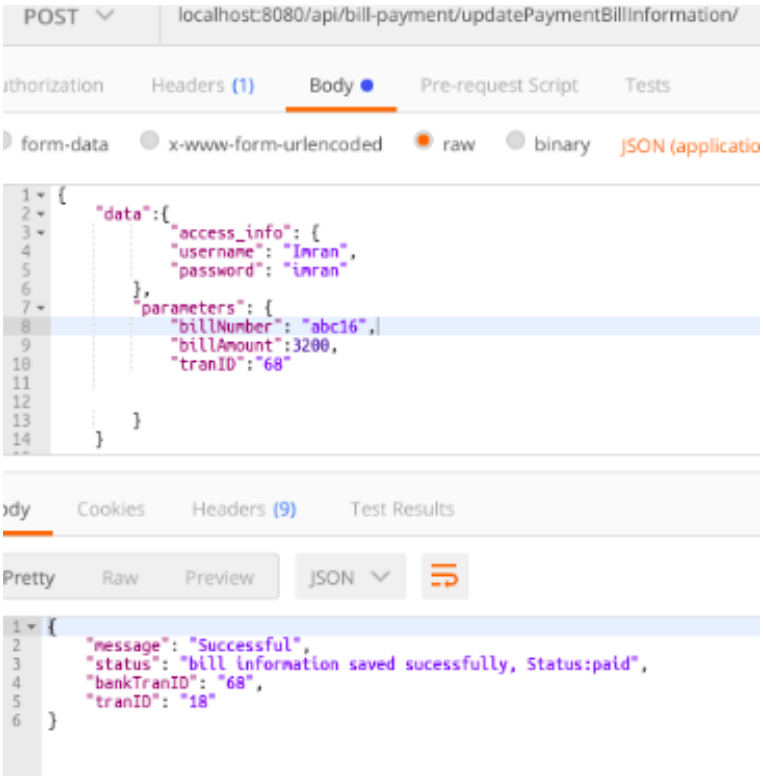

Figure 4.2: Update pay Bill Information

3. Get All Unpaind bills

request parameters:

customerNumber,userName,Password.

response data:

all data from billInfo table against given customerNumber

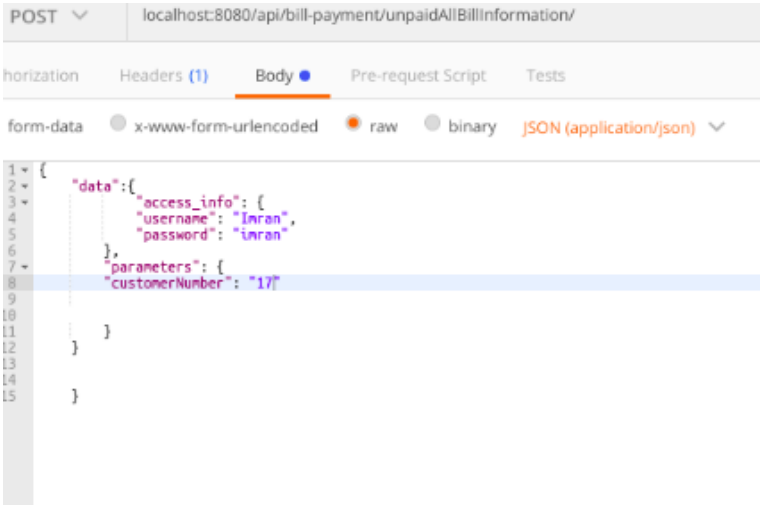

Figure 4.3: Unpaid bill input

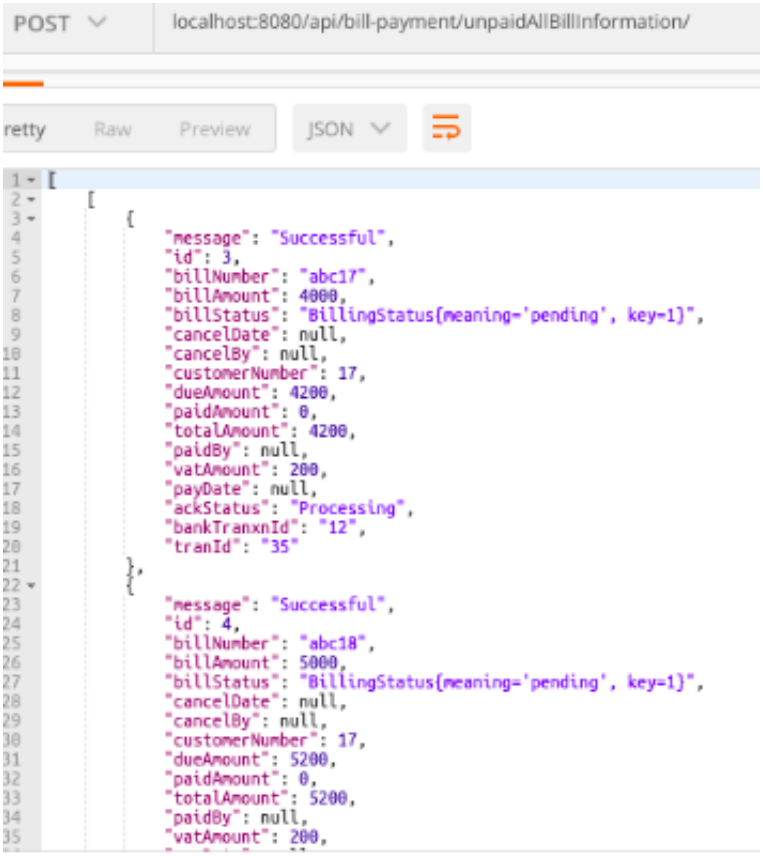

Figure 4.4: Unpaid bill Output

4. Bill Cancel API

request parameters:

billNumber, cancelRemarks,userName,Password.

Update billInfo table accordingly

response data:

status, message

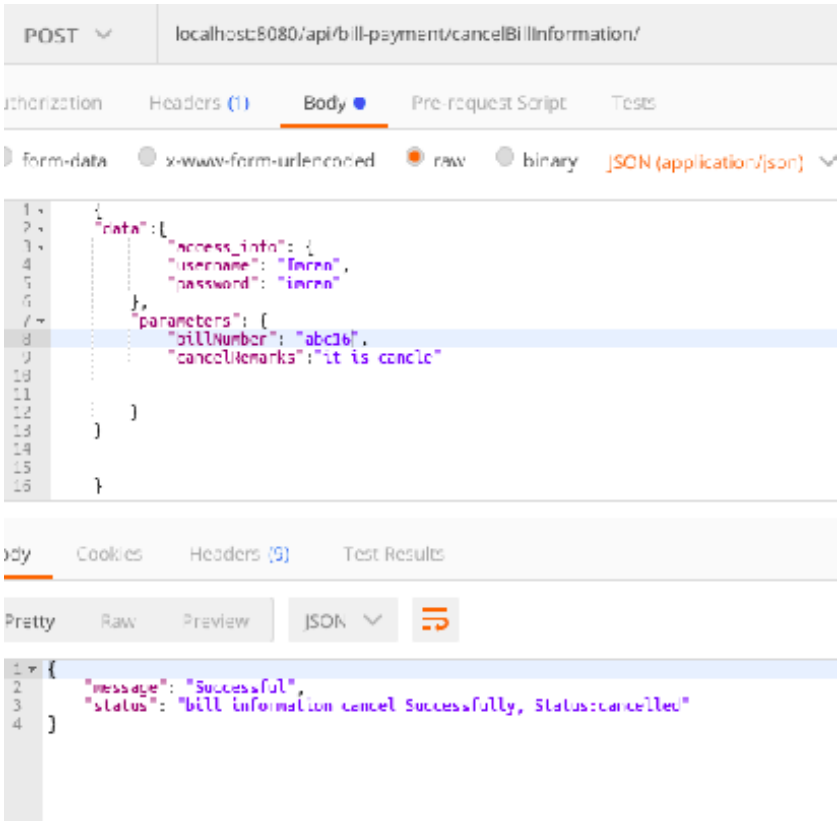

Figure 4.5: cancel bill Information

5. Get All Current Date bills

request parameters:

userName,Password.

response data: all data from billInfo table against current date.

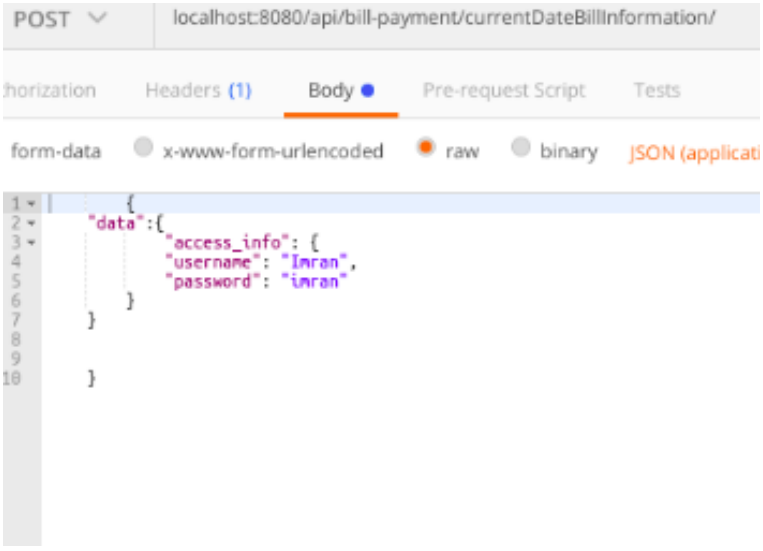

Figure 4.6: Current date input

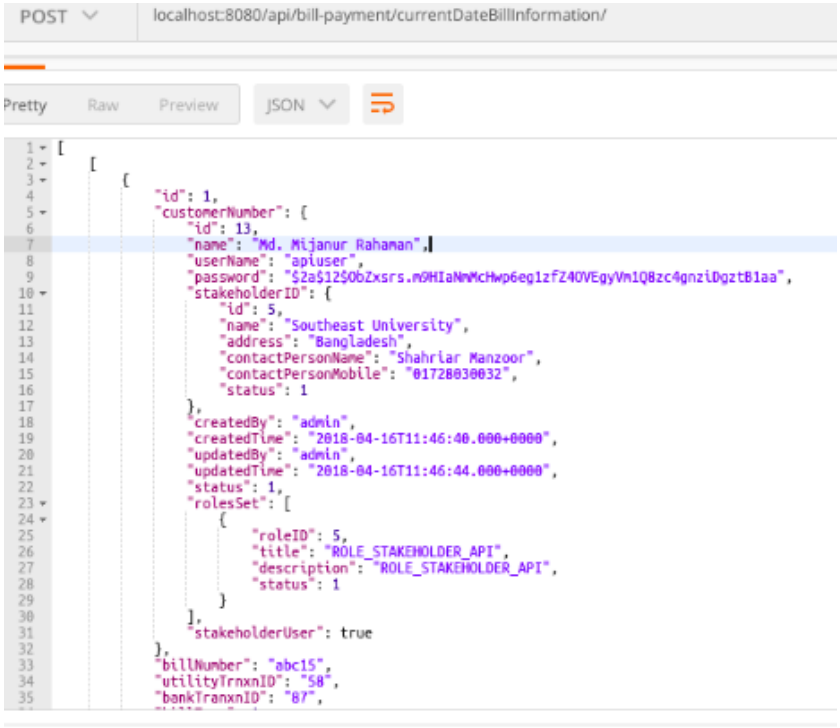

Figure 4.7: Current date output

6. Acknowledge Api

Add ackStatus column in billInfo table (default: "Processing")

parameters:

bankTranxnId, tranId, username, password

response:

if exists "OK" else "NOT OK" update ackStatus to "Acknowledged"/"Processing"

Figure 4.8: Acknowledgement

7. Call Back Api

if not acknowledged within 15 minutes of payment time http request to client Api(dummy)

parameters:

bankTranxnId, tranId, username, password

update ackStatus to "Call back sent"

Figure 4.9: call back to client

8. Reconciliation Api

date and type(summary/details) will be added in parameters section.

response(json) should contain following data:

if type is summary:

totalCount

totalAmount

totalSuccessCount

totalSuccessAmount

totalFailedCount

totalFailedAmount

if type is details:

totalCount, totalAmount, totalSuccessCount, totalSuccessAmount, totalFailed-Count, totalFailedAmount, bill data in array

response data:

username , password ,date ,type

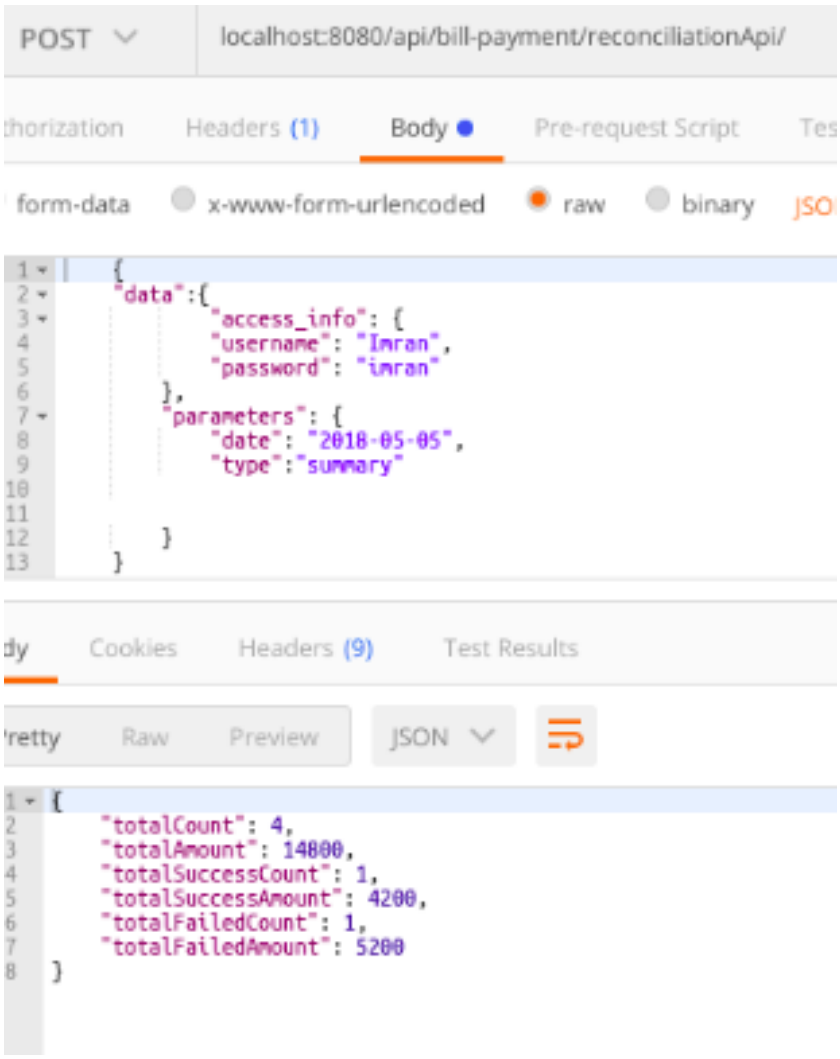

Figure 4.10: Reconcilation Summary

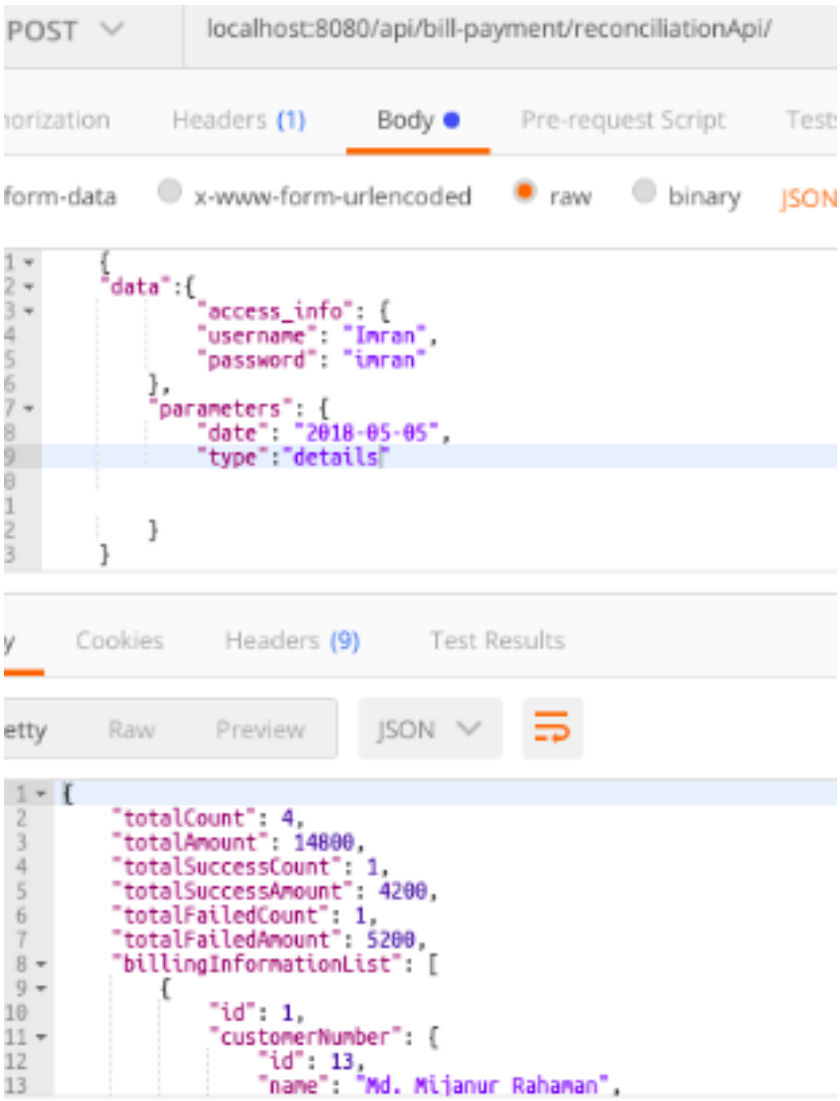

Figure 4.11: Reconcilation Details input

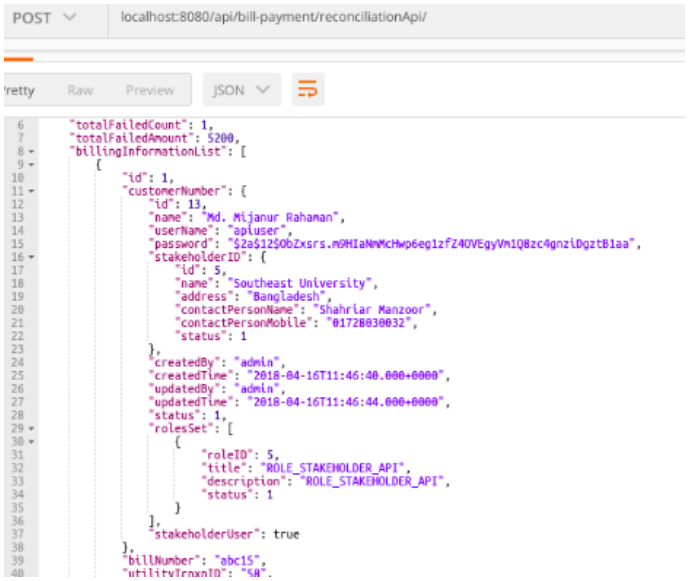

Figure 4.12: Reconcilation Details output

Chapter 5

System

5.1 Use Case Diagram

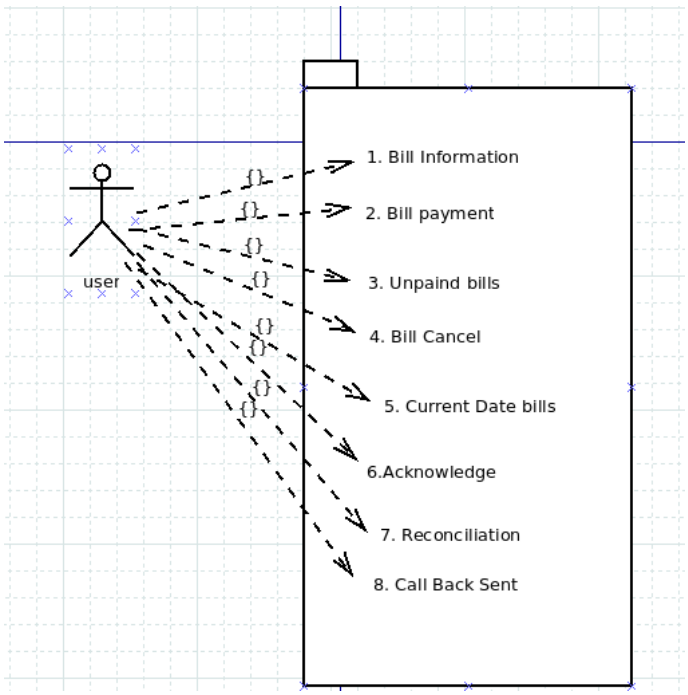

Figure 5.1: Use case Diagram

5.2 Sequence Diagram

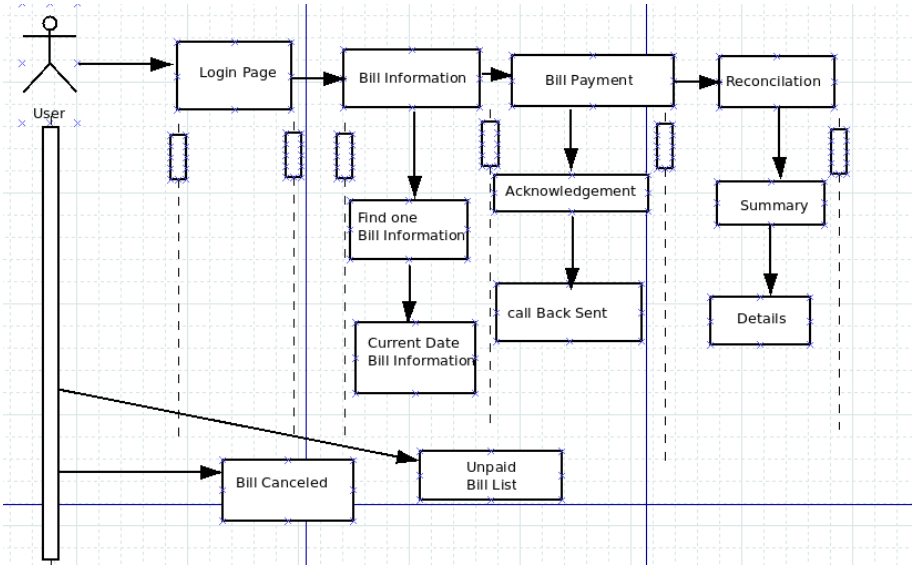

Figure 5.2: Sequence Diagram

5.3 Class Diagram

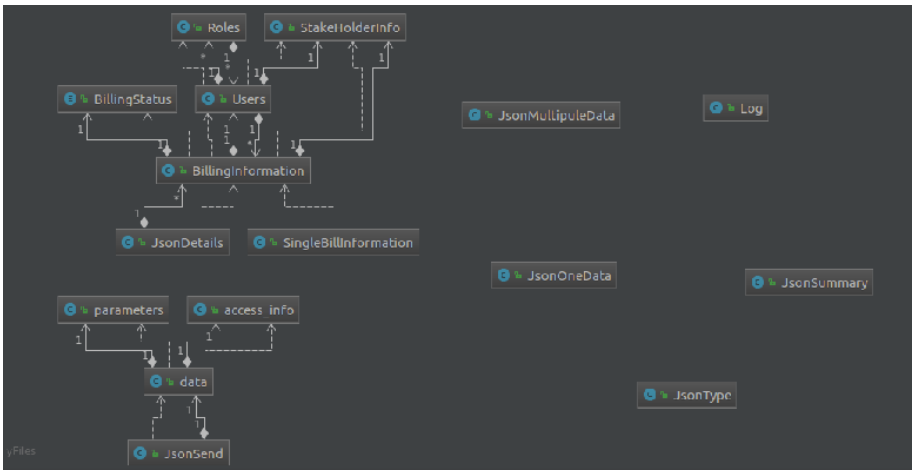

Figure 5.3: class Diagram

Chapter 6

Conclusion

In completely this internship was useful and successful exprience for me.I have gained new knowledge,skills and met many new people.I got insight into professional practice.I learned the different facets of working with this company.As a fresher I worked in a new team,It was a very challenging part for me to perform.The internship was also good to find out what is my weakness and strengths part.My co-supervisor to define what skills and knowledge I have to improve.However I could performe certain tasks in project better if I pratice know more project applied in certain studies.It would also be better if I can Present and express myself and more confidently.At last this internship course has given men new exprience.

Glossary

I worte this internship paper by using latex.So there have some limitation.I can't use some symbol like apostrophe , quotation mark etc.At first i faced some problem to use letex.Then I learn about how to use latex.I try to write all of my Internship work on this book.I try to write all details about my project which I try to finish in my internship.

Appendix A

This report wite based on Internship. Here include how , why , where and when i completed this work. My internship to my work is the field of web development.Web development is and important of software engineering.I try to finish my internship report according to my work.

Bibliography

[1] Title = Spring
URL = https://spring.io/

[2] Title = RESTful API
URL = http://www.restapitutorial.com/

[3] Title = Mysql
URL = https://www.mysql.com/

[4] Title = Spring RESTful API
URL = https://spring.io/guides/gs/rest-service/

[5] Title = Project Link
URL = hgit clone https://imranKhan2018@bitbucket.org/imranKhan2018/billpaymentapi.git