**MORGAN STANLEY 6 MONTHS INTERNSHIP**

**TEST INFRASTRUCTURE**

**INTERNSHIP REPORT**

*Submitted by*

PRABHAV SRIVASTAVA [Reg No:RA1811030010072]

*Under the Guidance of*

Dr. S. PRABAKERAN, B.Tech, M.E, PhD

(Assistant Professor, Department of Networking and Communications)

*in partial fulfillment of the requirements for the degree of*

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE ENGINEERING

WITH SPECIALIZATION IN CYBER SECURITY



DEPARTMENT OF NETWORKING AND COMMUNICATIONS

COLLEGE OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR- 603 203

MAY 2022



# SRM INSTITUTE OF SCIENCE AND TECHNOLOGY KATTANKULATHUR – 603 203

## BONAFIDE CERTIFICATE

Certified that this B.Tech project report titled “**MORGAN STANLEY 6 MONTHS INTERNSHIP REPORT**”is the bonafide work of **Mr. PRABHAV SRIVASTAVA** who carried out the project work under my/our supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion for this or any other candidate.

**SIGNATURE SIGNATURE**

##### DR. S. PRABAKERAN DR. ANNAPURANI ANAIYAPPA.K

**ASSISTANT PROFESSOR HEAD OF THE DEPARTMENT**

Department of Networking and Communications Department of Networking and Communications

**Signature of the Internal Examiner Signature of the External Examiner**

Department of Networking and Communications

##### SRM Institute of Science & Technology

##### Own Work\* Declaration Form

This sheet must be filled in (each box ticked to show that the condition has been met). It must be signed and dated along with your student registration number and included with all assignments you submit – work will not be marked unless this is done.To be completed by the student for all assessments

**Degree/ Course :** B.Tech Computer Science Engineering Cyber Security

**Student Name :** Prabhav Srivastava

**Registration Number :** RA1811030010072

**Title of Work :** Internship Review Morgan Stanley

I / We hereby certify that this assessment compiles with the University’s Rules and Regulations relating to Academic misconduct and plagiarism\*\*, as listed in the University Website, Regulations, and the Education Committee guidelines.

I / We confirm that all the work contained in this assessment is my / our own except where indicated, and that I / We have met the following conditions:

• Clearly references / listed all sources as appropriate

• Referenced and put in inverted commas all quoted text (from books, web, etc)

• Given the sources of all pictures, data etc. that are not my own

• Not made any use of the report(s) or essay(s) of any other student(s) either past or present • Acknowledged in appropriate places any help that I have received from others (e.g. fellow students, technicians, statisticians, external sources)

• Compiled with any other plagiarism criteria specified in the Course handbook / University website

I understand that any false claim for this work will be penalized in accordance with the University policies and regulations.

|  |
| --- |
| **DECLARATION:** |
| I am aware of and understand the University’s policy on Academic misconduct and plagiarism and I certify that this assessment is my / our own work, except where indicated by referring, and that I have followed the good academic practices noted above. |
| If you are working in a group, please write your registration numbers and sign with the date for every student in your group. |

**ACKNOWLEDGEMENT**

We express our humble gratitude to Dr C. Muthamizhchelvan, Vice-Chancellor, SRM Institute of Science and Technology, for the facilities extended for the project work and his continued support. We extend our sincere thanks to Dean-CET, SRM Institute of Science and Technology, Dr T.V.Gopal, for his invaluable support. We wish to thank Dr Revathi Venkataraman, Professor , Chairperson, School of Computing, SRM Institute of Science and Technology, for her support throughout the project work.

We are incredibly grateful to our Head of the Department, Dr K. Annapurani Panaiyappan Professor, Department of Networking and Communications, SRM Institute of Science and Technology, for her suggestions and encouragement at all the stages of the project work. We want to convey our thanks to our Panel Head, Dr.S.Prabakeran, B.Tech, M.E, PhD Assistant Professor, School of Computing Department of Networking and Communications SRM Institute of Science & Technology Kattankulathur, Chengalpattu District - 603 203 Tamil Nadu, for their inputs during the project reviews and support. We register our immeasurable thanks to our Faculty Advisor, Dr. Uma , Software Department, SRM Institute of Science and Technology, for leading and helping us to complete our course. Our inexpressible respect and thanks to my guide, Dr. Prabhakaran, Professor , Networking Department , SRM Institute of Science and Technology, for providing me with an opportunity to pursue my project under Morgan Stanley. He provided me with the freedom and support to explore the research topics of my interest. His passion for solving problems and making a difference in the world has always been inspiring. We sincerely thank the Software Department staff and students, SRM Institute of Science and Technology, for their help during our project. Finally, we would like to thank parents, family members, and friends for their unconditional love, constant support, and encouragement.

Prabhav Srivastava

**ABSTRACT**

Training and developing the right skill set to deliver high-quality software, enhance my skills and familiarize myself with new Technologies over a time frame of 6 months. Working on real life projects, building a test infrastructure, creating new features and developing the software under the guidance of my Manager , Buddy and Team.

Main target of the internship is to work on and build a testing infrastructure software that will support multiple modules that the CVA team is going to use on a day to day basis solving many issues and removing multiple iterative processes that will eventually speed up the development process and save multiple work hours and increase productivity drastically.

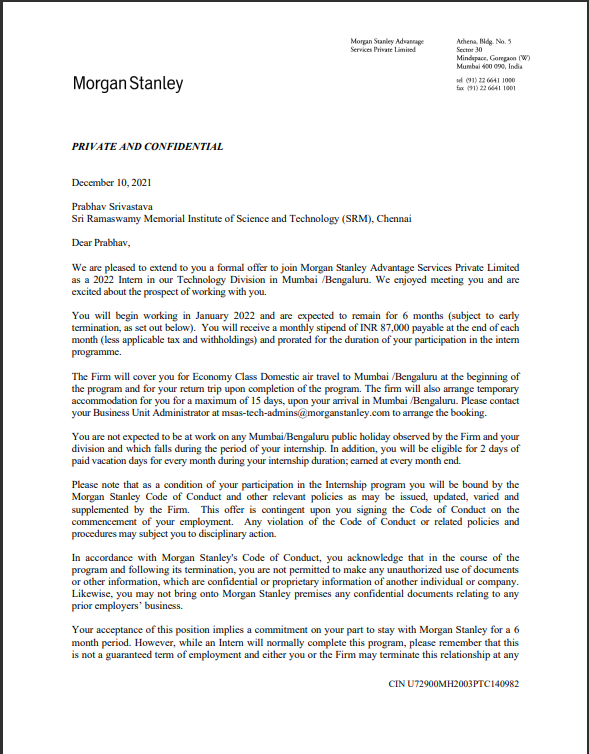
The test infrastructure has 4 modules to it.

* DB2 Testing
* EWM Testing
* T0 Testing
* Grid Testing

Morgan Stanley has multiple domains such as Wealth Management (WM), (IST), (AI) etc. Under IST Division there is the CVA Team. This project is being built for the CVA team that has half the team based in India spread between Mumbai and India and the other half in the US, New York. The application is being built for the employees.

Will be working as a developer on the forthcoming projects and will be adding features as well as building applications for the Team.

**OFFER LETTER**

****

**TECH STACK**

* Angular
* HTML
* CSS
* JavaScript / Typescript
* Java
* Spring Boot
* REST API’s
* UNIX Environment
* Shell Scripting

**TABLE OF CONTENT**

|  |  |  |
| --- | --- | --- |
| **SERIAL NUMBER** | **TITLE** | **PAGE NUMBER** |
| 1. | Cover Page and Title | 1 |
| 2. | Bonafide Certificate | 3 |
| 3. | Own Work Declaration | 4 |
| 4. | Acknowledgement | 5 |
| 5. | Abstract | 7 |
| 6. | Chapter 1 : Introduction | 9 |
| 7. | Chapter 2 : Related work | 12 |
| 8. | Chapter 3 : System Demo | 19 |
| 9. | Chapter 4 : Modules | 27 |
| 10. | Chapter 5 : Morgan Stanley Training | 33 |
| 11. | References | 44 |

**TABLE OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **CHAPTER NUMBER** | **TITLE** | **PAGE NUMBER** |
| 1 | SDLC | 14 |
| 2 | HOME PAGE APPLICATION | 19 |
| 3 | DB2 TESTING | 21 |
| 4 | GIT STATIC TABLES | 23 |
| 5 | WORKFLOW STATIC TABLES | 24 |
| 6 | RUNWAY LOGS | 25 |
| 7 | RUNWAY SCHEMA WISE | 26 |
| 8 | SLICE DATA LOADING | 28 |
| 9 | JAVA SPRINGBOOT SERVER | 29 |
| 10 | UNIX ENV | 29 |

**CHAPTER 1**

**INTRODUCTION**

**1.1 DOMAIN OF THE PROJECT**

The domain of this project falls under the major Category of Software Development Engineering (SDE) and the sub domain of the same would be Application Development (AD). The application being built will be a Full Stack Application that has the frontend and backend being handled from scratch.

FE Technology : Angular, Typescript, HTML CSS JS

BE Technology : Java and Spring Boot

**1.2 PROJECT**

Building a cross platform browser application from scratch that will potentially change the whole system of using multiple PR’s and dependant testing and will digitalize the process and at the same time will automate the testing process giving the developers a lot more flexibility and surety that the code and files being pushed into the build will be accurately and thoroughly tested. Will build the application in such a way that even outside organizations in Morgan Stanley will be able to use the application according to their requirement with minor changes in the code base.

**1.3 MOTIVATION**

The issue of the wait time and the uncertainty it causes is a big issue and that is something that everyone from our team faces at one point or another.

Being able to solve this issue will give me immense pleasure and satisfaction as with the help of my hard work and education at SRM , I was able to solve a real life problem faced by our company and was able to contribute back to the team.

There were a few problems with the system identified by my Manager Mr. Somesh Bansal (Vice President , Morgan Stanley). The current system will benefit from the project being developed right now as the segregation of the multiple modules and addition of the individual testing functionality provided by the application will make the development process faster and more accurate and error free making production ready changes available as soon as possible thus saving a lot of working hours and these hours in turn can be used in developing multiple features

**1.4 PROBLEMS FACED BY THE CURRENT TEAM**

* Uncertainty of code accuracy being pushed
* No testing infrastructure for individual testing
* No middle man between dev user and master code management user
* Loss of time waiting for PR’s to be resolved
* Swift process unavailable.

**1.5 CHALLENGES FACED DURING THE DEVELOPMENT**

* Integration of the 4 modules
* Accessing the Morgan Stanley Dev env through local computers
* System Design
* Windows - UNIX transfer
* Learning new technologies

**CHAPTER 2**

**RELATED WORK**

**2.1 LITERATURE SURVEY**

At first the company had provided us with the number of courses that we had to complete from the pluralsight website alongside 50 days of hands-on training in various domains and multiple technologies taken by SMEs from the firm itself. The problem statement was given after the training sessions had taken place and then the work towards the same started thereafter.

The problem statement was broken down into 4 modules and each module was to be handled separately.

* + DB2 Testing
  + EWM Testing
  + T0 Testing
  + Grid Testing

A thorough deep dive into the topics were done and the designing phase was put into action. Once the designing phase was complete the technologies were put up on vote , the decision of what tech stack was best suited for the given probl;em statement was then done and inferences were made. The weekly targets were set and the project work started on the go.

**2.2 INFERENCE OF THE LITERATURE SURVEY**

* The design of the application was set up and how we were going to handle the backend and frontend part of the application was set to stone.
* Tech stack was finalized and the setting up of the project started based on the inferred tech stack (Angular, HTML, CSS, Javascript & Typescript,
* Java, Spring Boot, REST Api’s, UNIX Environment, Shell Scripting)
* Meeting with the higher ups was set and the project idea was tossed towards them , feedback was taken and the work started.
* The frontend was to be started with first off.

**2.3 SDLC**



**ANALYSIS AND PLANNING NEEDED**

* Needs analysis is a very important and fundamental phase in the SDLC. It is made up of senior team members with ideas from customer, sales department, market surveys and industry experts in the industry. This information is then used to plan the basic project approach and to conduct feasibility studies in a cost-effective, operational and technological environment.
* Planning for quality assurance requirements and work-related risk identification are also done in the planning phase. Theiresult of research into the feasibilityuof a technical task is to defineythe different technical methods that can be followed to successfully implement a projectqwithout the slightest risk.

**DEFINING REQUIREMENTS**

* Once the demand analysis has been done the next step is to clearly define and document the product requirements and get them approved by the customer or market analysts. This is done through the SRS (Software Requirement Specification) document that contains all the product requirements that must be designed and developed during the life cycle of a project.

**DESIGNING PHASE OF THE PRODUCT**

* SRS is a guide for product designers to come up with the best designs for the product to be developed. Based on the requirements in S.R.S, the design of more than 1 product design is generally proposed and documented in the DDS i.e the Design Document Design.
* This DDS is reviewed by all key stakeholders and is based on various parameters such as risk assessment, product durability, design consistency, budget and time constraints, the best design method for product selection.
* The design approach clearly defines all product structure modules and their interactions with the data flow representation of external and external modules (if any). The internal design of all proposed building modules should be clearly defined in the smallest detail in the DDS.

**BUILDING AND DEVELOPING THE PRODUCT**

* At this stage in theiSDLC real development begins and theiproduct is built. The edit code ishgenerated as per DDS for this section. If the designhis done in a detailed andgsystematic way, code production can be achieved without much hassle.
* Engineers should follow the coding guidelines defined by their organization as well as editing tools such as editors, interpreters, debuggers, etc. used to generate code. Advanced editing languages as C/C ++, Pascal, Java(JVM) and PHP are used for coding. The programming language is selected according to the type of software being developed.

**PRODUCT EVOLUTION**

* This category is usually the subset of all categories as in modern SDLC models, testing activities are more involved in all categories of SDLC. However, this category refers to the product only testing phase where product damage is reported and tracked, corrected and re-evaluated, until the product meets the quality standards set out in SRS.
* The unit testing starts out here before market delivery, the product is run through standard test cases to assure build quality and edge cases to check whether it can withstand all scenarios successfully.

**MARKET DELIVERY**

* Once the product has been inspected and readyafor shipment it is officially released from the appropriate marketplace. Sometimesaproduct delivery occurs ina stages according to the bizastrategy of that organization. The product may first be released in aalimited edition and tested in a s real business environment (UAT User acceptance test).
* Then based on feedback, the product may be released as is or through enhanced features in the target market segment. After the product is released to the market, its maintenance is carried out on an existing customer base.

**SDLC MODALS**

* There are software life cycle models described and designed that are followed during software development. These models are also called Software Development Process Models.

The following are some of the most important and well-known SDLC models in the industry

* WaterfallaModel
* RepeataModel
* WindingaModel
* V-modela
* Big Bang models
* Other compatible alts are Agile Model, RAD Model, Rapid App Dev and Prototyping Mode

**CHAPTER 3**

**SYSTEM REQUIREMENTS**

**3.1 PROBLEM DEFINITION**

Main target of the internship is to work on and build a testing infrastructure software that will support multiple modules that the CVA team is going to use on a day to day basis solving many issues and removing multiple iterative processes that will eventually speed up the development process and save multiple work hours and increase productivity drastically.

The test infrastructure has 4 modules to it.

* DB2 Testing
* EWM Testing
* T0 Testing
* Grid Testing

Morgan Stanley has multiple domains such as Wealth Management (WM), (IST), (AI) etc. Under IST Division there is the CVA Team. This project is being built for the CVA team that has half the team based off in India spread between Mumbai and India and the other half in the US, New York. The application is being built for the employees in these 3 regions.

**3.2 PROPOSED METHODOLOGY**

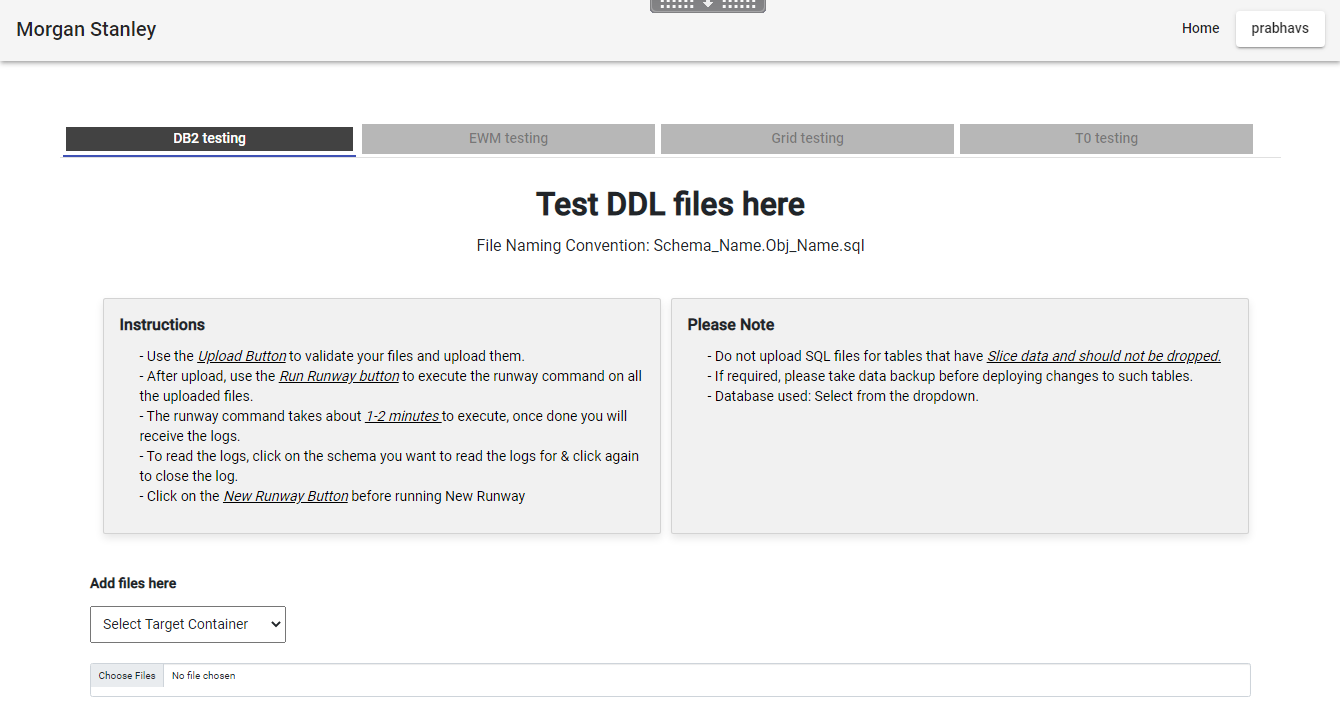
A thorough deep dive into the topics were done and the designing phase was put into action. Once the designing phase was complete the technologies were put up on vote , the decision of what tech stack was best suited for the given probl;em statement was then done and inferences were made. The weekly targets were set and the project work started on the go.

**3.3 VIRTUAL DEMO**

Due to the company NDA policy , the Use case , System design and Any paper / Video with respect to the project cannot be disclosed.

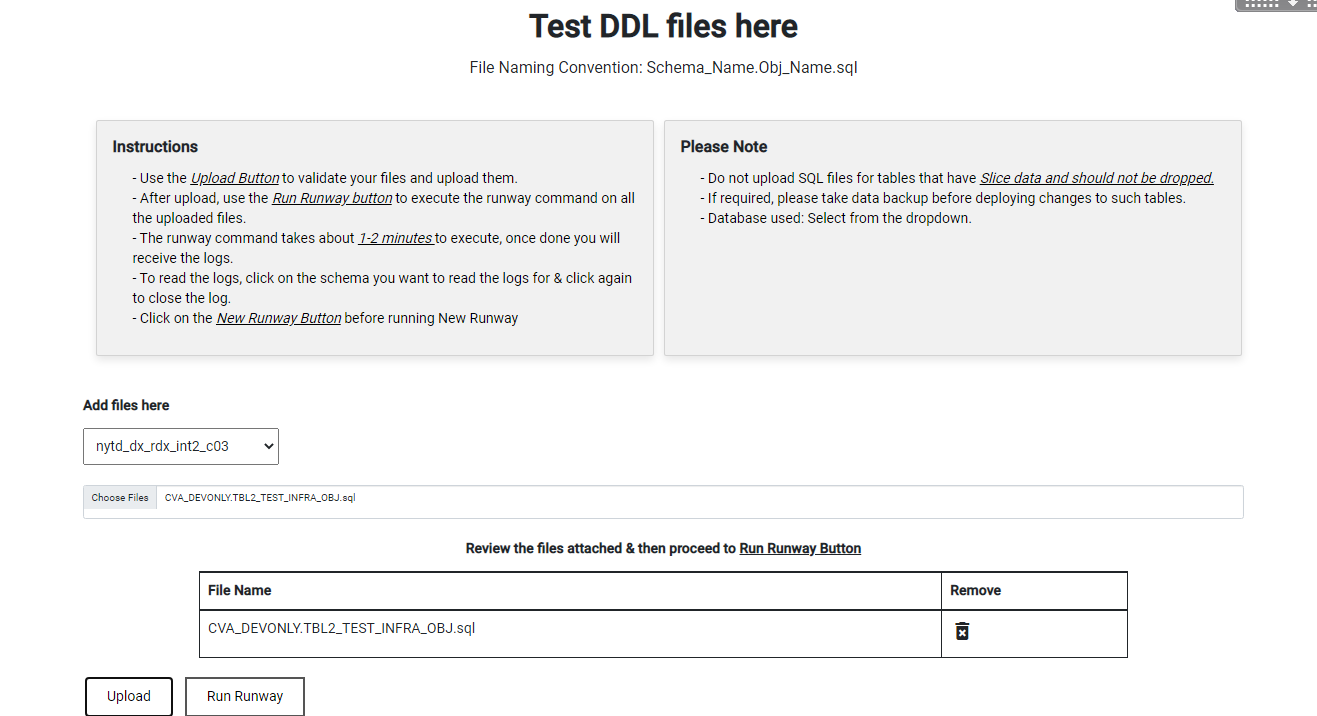
Here is a Virtual Demonstration of the project with the help of images.

1. Home Page of the Application



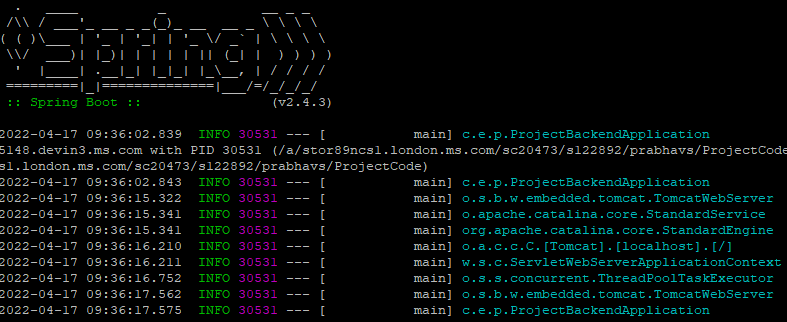
1. DB2 Testing

* We select the delphix container
* We select the ddl files that are required to test using the runway command
* We upload the selected ddl files to the UNIX environment from our respective systems
* We then test using the Run Runway command button.



1. Spring Boot server Running in the background

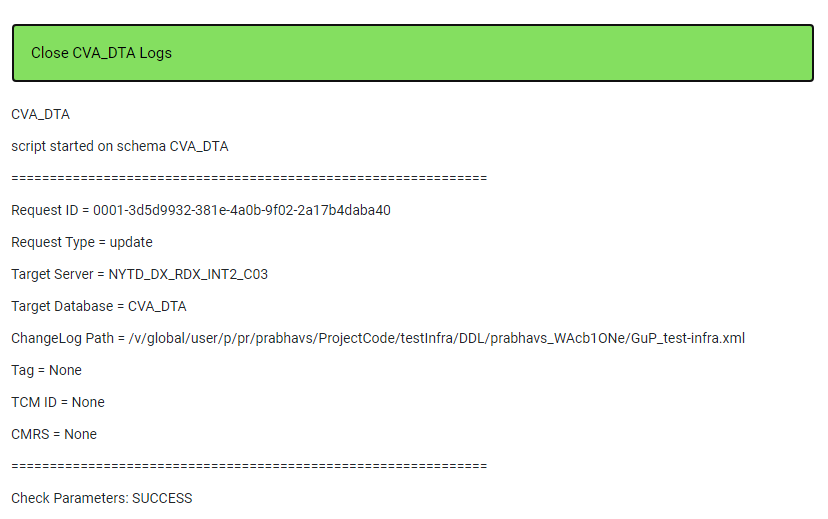
* Takes care of the entire backend part of our application
* Its is the java jar file that is running in the UNIX environment.

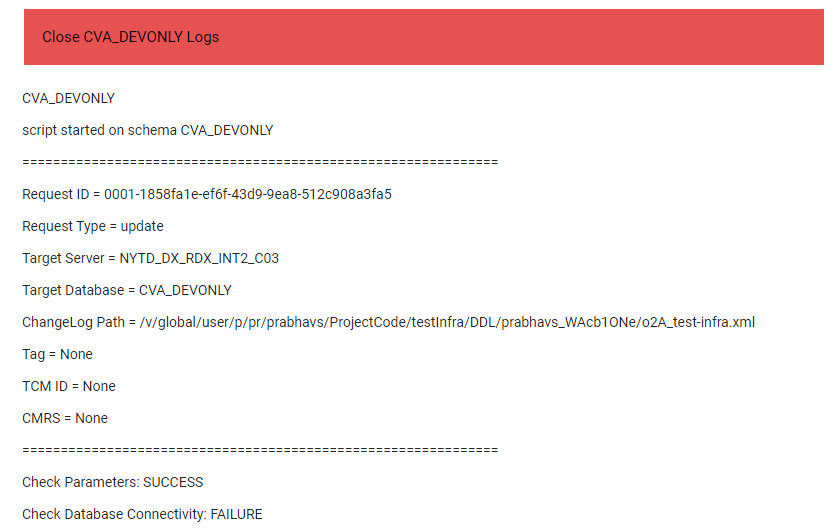


1. Runway command runs

* Once the runway command runs on the selected files, the output is given
* Red means theres an error in the file
* Green means there is no error in the file and that this can be pushed into the main container



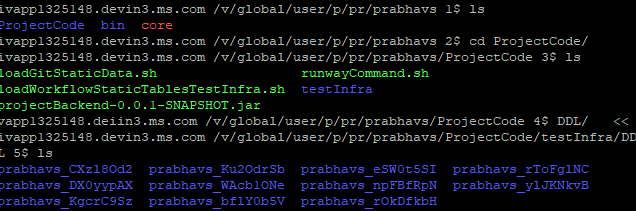


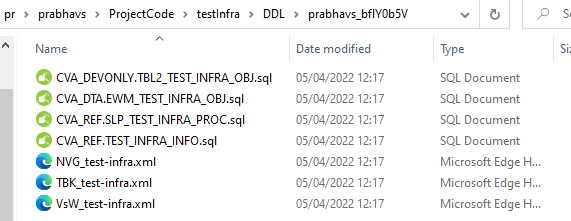


Using These logs, the user can figure out where the error lies and how to go about it to fix this.

**4.** Backend Side of the Runway

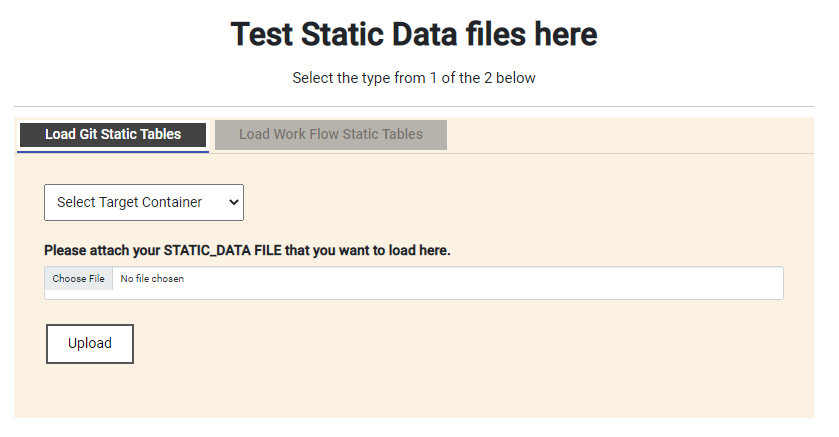
* The files are stored in the backend algorithmically

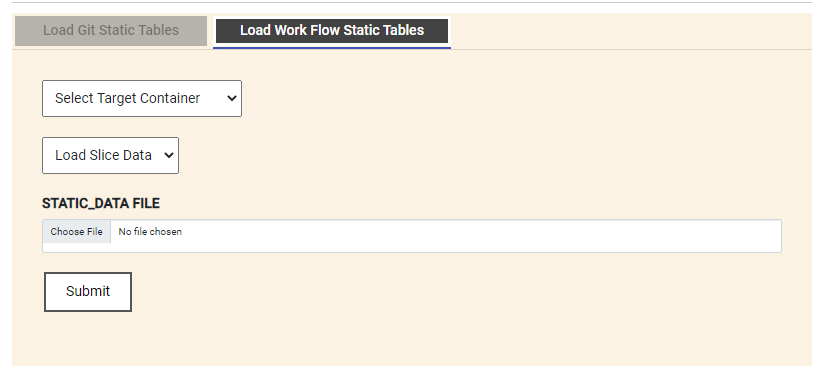


-

1. EWM Backend Side

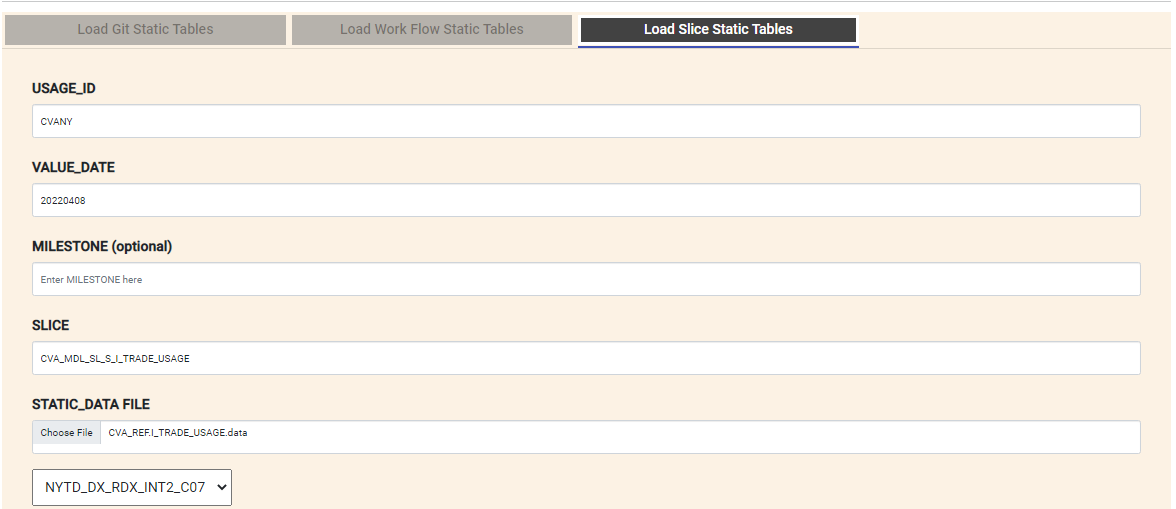
* EWM : Event Workflow Management
* Static Data Files are attached and their testing is done in this module
* There are 2 sub modules within this module





1. Slice Data Interactive Services

* Slice Static Data Loading is done Via the interactive Services
* Here the Required fields are added and a input.json is generated and sent to the UNIX environment for processing
* Here Scripts make call to the services and the execution takes place.



**3.4 SOFTWARE AND HARDWARE COMPONENTS**

Manufacturer : Lenovo

Model : Ideapad 320

System Type : 64-bit Operating System

Operating System : Windows 10

Processor Type : Intel Coree i5 7th Gen

Processor Speed : 3.20Ghz

Installed Memory : 8 GB

Number of Cores : 8

Number of Threads : 16

Other tools : VS Code, Intellij , File Manager, PuTTY, Chrome, UNIX

**CHAPTER 4**

**MODULES**

**4.2.1 DATA PROCESSING**

i) Why is data processing important?

Pre-data processing is primarily a test of data types. As we will be making API Calls we want the data to be validated and checked before hand so that no wasteful calls are made eventually bringing down the efficiency of the application thus making it marginally slow.

* + - Accuracy: Checking whether the data entered is correct or not.
    - Completeness: Checking whether data is available or not.
    - Consistency: To check that the same data is stored in all the same or different locations.
    - Time: Data should be updated appropriately.
    - Reliability: Data must be reliable.
    - Interpretation: Data comprehension.

1. Major Data Processing Before API Calls:
   * Cleaning data
   * Data reduction
   * Data processing
   * Data Validation
   * Data Type Checking
   * File delimiter checks

iii) Data aggregation:

The process of combining multiple sources into a single database. The Data Integration Process is one of the key components of data management. There are some issues to consider during data consolidation.

iv) Schema integration:

Incorporates metadata (a set of data that describes other data) from different sources. Business identification problem: Identifying businesses from multiple databases. For example, a system or application should know \_id a student of one website and the name\_ of a student on another website belongs to the same business.

Finding and solving data value concepts: Data taken from different websites while compiling may be different. Like the number of attributes from one website may be different from another website. For example, the date format may vary as "MM / DD / YYYY" or "DD / MM / YYYY".

**4.2.2 USER CLASSIFICATION**

We will be dividing the system users into 2 categories, the employee & the administrator. Each user will have separate privileges and access to certain sections of the application. The admin will have authority to create and manage the code base.

**4.3 AGILE METHODOLOGY**

**4.3.1 WHAT IS AGILE?**

Agile is the ability to create and respond to change. It is a way of coping, and finally succeeding, in an uncertain and chaotic environment. The authors of the AgileqManifesto chose "Agile" as theqlabel for the whole idea because thataword represented thehflexibility and responsexto change that was so importantfto their approach.

It’s about thinking about how you can understand what’s going on in your area today, find out what uncertainty you are experiencing, and discover how you can adapt to that as you go along.

**4.3.2 WHAT IS AGILE SOFTWARE DEVELOPMENT?**

Fast software development goes beyond components such as Scrum, Extreme Programming, or Feature-Driven Development (FDD). Fast software development goes beyond practices such as paired editing, test-driven development, posture, editing times, and sprints. Agile software development is an umbrella term for a set of frameworks and processes based on the values ​​and principles set out in the Agile Software Development Manifesto and the 12 Principles behind it. When you approach a software development in a certain way, it is usually best to live by these principles and principles and apply them to help you find the right things to do when considering your particular context.

One thing that sets Agile apart from other software development methods is the focus on the people who do the work and the way they work together. Solutions arise from collaborations between different groups working on self-organizing using appropriate processes in their context. There is a strong focus on the Agile software development community in partnership with the planning team.That doesn't mean there are no bosses. It means that teams have the ability to see how they will deal with things themselves.

It means that those groups are different. Those teams do not have to have the same roles that you have, so when you put together a team, you make sure you have all the right skills in the team. There is still a place for management. Managers make sure team members have, or acquire, the right skills. Managers provide an,environment that allowsithe team to succeed. Managers oftennback off and allow their team to see how they will deliver the products, but they intervene when teams try but are unable to solve problems.

When more teams and organizations start doing Agile development, they focus on actions that help in collaboration and project planning, which is a good thing. However, another key set of procedures that can always be followed but must be specific technical procedures that directly address software development in a way that helps your team deal with uncertainty. Those technical procedures are important and something you should not ignore.

**4.3.3 AGILE KEY CONCEPTS**

Below are a few key Agile concepts. You can see more in our wording section.

**User News:** In consultation with the customer or product owner, the team classifies the work to be done with an additional functionality called “user issues.” Each user story is expected to contribute to the overall value of the product.

**Daily Meeting**: Each day at the same time, the team meets to give everyone the information they need to get involved: each team member briefly describes any “completed” donations and any obstacles that hinder them.

**People:** When a project needs - for example when user information is a key factor in project results - the team creates a lot of details, live transcripts of fake users of future products: these are called "personas."

**Group:** The “team” in the Agile sense is a small group of people, assigned to the same task or effort, almost all of them full-time. A number of team members may be temporary participants, or have competing responsibilities.

**Improved Development:** Almost all Agile teams love the evolving development strategy; in the Agile context, this means that each successive version of the product is usable, and each builds on the previous version by adding functionality to the user.

**Repetitive Improvements:** Fast projects are repetitive as they deliberately allow "repetition" of software development tasks, as well as the opportunity to "revisit" the same work products.

**Milestone Retrospective:** When a project has been in progress for a long time, or at the end of a project, all permanent team members (not just developers) invest from one to three days in detailed analysis of key project events.

**CHAPTER 5**

**MORGAN STANLEY TRAINING**

1. **OS FUNDAMENTALS**

Started off with a Q n A session going over the basic topics and basic questions related to Operating Systems and everything about them we have learnt during the course of 4 years of out B.Tech Program. The Further technical sessions involved deep sessions and discussions going over Process , Thread and their essentiality.During the day to day running of the firm, how it is used in application development and the importance of thread safety.

Day 2 Session presided over fundamentals

* Application software
* System software
* Types of operating systems
* Multiprogramming
* Multitasking
* Duties of an operating system

Day 3 Process Scheduling

* Process
* Process Scheduling
* Process states
* Context switching
* Thread

1. **DATA STRUCTURES**

Data Structures The Data Structures (DS) course introduces the basic and advanced concepts of data structures. The Data Building course is designed for beginners and experts. A data warehouse is a way to store and organize data for efficient use. The study of data structures covers all data structure topics such as arrays, identifiers, structures, linked lists, stacks, lines, graphs, searches, filtering, programs, and more.

Numerous ways to organize data/input/output in memory. Array is a collection of memory elements where data is stored sequentially, that is, one after another. In other words, we can say that the list keeps the elements continuous. This data organization is done with the help of data structures. There are other ways to organize data in memory. Let's look at the different types of data structures.

The data structure is not any programming language such as C, C ++, java, etc. A set of algorithms that can be used in any programming language to process data in memory. To organize data in memory, a 'n' number of algorithms was proposed, and all of these algorithms are known as Abstract data types. These types of invisible data are a set of rules, Data Structures Study

Data Structure Types

There are two types:

* Original DS
* Non-classic DS

Initial data structure

Old data structures are a type of outdated data. Int, char, float, double, and ptr are old DS that can hold single values. Formatting Non-Original Data There are two types of data structures.archives. String Data Structures Indirect Data Structures Linear Data Structures

A sequential order of data is called a linear data structure. The data structures used for this purpose are arrays, linked lists, stacks, and strings

Non-Original Data Formatics

The archive data structure is divided into two types:

* Line dataastructure
* Indirect dataastructure
* Linear DataaStructure

A sequential data structure is known as a straightforward data structure. The data structures used for this purpose are Arrays, Link List, Stacks, and Line. In these data structures, one object is connected only to one component by a line.

When one element is connected to the 'n' number of objects known as an indirect data structure. The best example of all trees and graphs. In this case, the elements are organized in an orderly fashion.

Data structures can also be categorized as:

* Stable data structure: It is a type of dataastructure where sizeaat the time of consolidational. Therefore, the max.size() is constants.a
* Powerful data structure: It’s a category of DS where size() is allocated at launch timings. There fore the large size is flexy.

**PURPOSE OF A DATA STRUCTURE**

* Searchinga: We search for anything in the DS.
* Filterings: We filter partof the DS in an ascending and then in the descending order.
* Installations: We lso add a new feature to the data structure.
* Reviews: We can also update the element, that is, we can replace the element.
* Deletions: We can also perform a deletions task to remove a feature from the DS

**ANGULAR TRAINING**

Angular: A platform and framework for building single page client apps using HTML CSS and TS. Angular is written in TS. It uses a set of key features and optional TS libraries that are imported into the application. Angular app structure is based on specific key concepts. The basic building blocks of the Angular Framework are the Angular sections arranged in Modules.

The Modules compile related codes into functional sets; Angular application defined by a set of NgModules. The app always has a root module that allows bootstrapping, and usually has multiple feature modules. The sections define the view, which is a set of screen objects Angular to choose from and transform according to the rationality of your system and data. Utilities use elements, which provide certain functions that are not directly related to viewing. Service providers can be injected into components such as dependence, which makes your code module, usable, and efficient. Modules

Components and services classes use decorators. The section metadata is linked to a template that describes the view. The template includes standard HTML and Angular guides as well as binding cuffs that allow Angular to modify HTML before rendering it for display.

Service class metadata provides the information Angular requires to make it available to components by dependency injection (DI).



Angular Modules

Are separate and compatible with JavaScript modules , NgModule announces the integration context of a set of components assigned to an application domain, workflow, or set of related skills. NgModule can integrate its components with related code, such as services, to create functional units.

Every Angular app has a root module, commonly called AppModule, which provides bootstrap that launches the application. The application contains many active modules.

Like JavaScript modules, NgModules can import the functionality of other Modules, and allow their functionality to be exported and used by other Modules. For example, in order to use the Route service in your application, you must import the Route Module

Every Angular application has at least 1 component which is called and named as the root component that connects to the component section and the page DOM model. Each section describes a class containing application plus the logical content i.e data, and is associated with an HTML template that describes the concept to be displayed in the target area.

The @ Component () template quickly identifies the class below it as a component, and provides a component template and metadata.

1. **JAVA SPRING BOOT**

Spring Boot Java Spring Framework (Spring Framework) is a popular, open source, business lev framework for creating standa lone, production based applications running Java Virtual Machine (JVM). Java Spring Boot (Spring Boot)

is a tool that enables the development of web applications and microservices with S pring Framework quickly and easily with 3 key skills: Automatic configuration A method with configuration ideas Ability to create standalone

applications These features work together to provide you with a tool that allows you to set up a Springbased app with minimal c onfiguration and setup. Why is the Spring Framework so popular? The Spring Framework provides a

dependency injection feature that allows items to define their dependence on which the Spring container is later installed. This allows developers to create modular applications that integrate freely co nnected components suitable

for microservices and distributed network applications. The Spring Framework also provides builtin support for standard functions that the app needs to perform, such as d ata binding, genre conversion, authentication, alternative management, resource management and events, internation al operations, and more. It includes various Java EE

Technologies such as RMI (Remote Method Invocation), AMQP (Advanced Message Queuing Protocol), Java Web Services, and more. Overall, the Spring Framework provides dev elopers with all the tools and features needed to create

free, compact Java EE applications running anywhere. What adds a Spring Boot to the Spring Framework ? As capable and comprehensive as the Spring Framework, it still requires significant time and knowledge to configur e, configure,

and implement spring applications. Spring Boot reduces this overhead in three important ways: I:

Autoconfiguration Autoconfiguration means that your application will run with a preset confidence that you don't need to manually configure the APP. Java Spring Boot has a built-in auto-stop feature, so it defaults to Spring Framework

Java Spring Framework (Spring Framework) is a popular, open source, business-level framework for creating standalone, production-based applications running Java Virtual Machine (JVM).

* Automatic configuration
* A method with configuration ideas
* Ability to create standalone applications

These features work together to provide you with a tool that allows you to set up a Spring-based app with minimal configuration and setup.

Why is the Spring Framework so popular?

The Spring Framework provides a dependency injection feature that allows items to define their dependence on which the Spring container is later installed. This allows developers to create modular applications that integrate freely connected components suitable for microservices and distributed network applications.

The Spring Framework also provides built-in support for standard functions that the app needs to perform, such as data binding, genre conversion, authentication, alternative management, resource management and events, international operations, and more. It includes various Java EEitechnologies such as RMI (Remote Method Invocation), AMQPi(Advanced Message Queuing Protocol), JavayWeb Services, and more. Overall, the Spring Framework provides developers with all the tools and features needed to create free, compact Java EE applications running anywhere.

What adds a Spring Boot to the Spring Framework ?

As capable and comprehensive as the Spring Framework, it still requires significant time and knowledge to configure, configure, and implement spring applications. Spring Boot reduces this effort with three important skills.

I: Automatic configuration

Automatic configuration means that applications are started with a pre-set reliance that you do not need to configure yourself. As the Java Spring Boot comes with built-in auto-stop capabilities, it automatically prepares both the basic Spring Framework framework and third-party packages based on your settings (and based on advanced processes, which helps avoid mistakes). Although you can override this default once the launch is complete, the Java Spring Boot configuration feature lets you start upgrading your Spring-based apps faster and reduces the chances of having human errors.

II: A visionary approach

Spring Boot uses a thoughtful approach to adding and setting beginner dependencies, based on the needs of your project. Following its judgment, Spring Boot decides which packages to include and which default prices to use, rather than requiring you to make all those decisions and set everything up manually.

You can define the requirements for your project during the startup process, when choosing between multiple startup dependencies — called spring launchers — that include standard operating conditions. You are using Spring Boot Initializr for completing a simple web form, without having to write a code.

For example, relying on the Spring Web Launcher allows you to build Spring-based web applications with minimal configuration by adding all

required dependencies, such as Apache Tomcat web server, to your project. "Spring Security" is another popular

executable dependency that automatically adds authentication and access control capabilities to your application. Spring Boot includes over 50 Spring Starters, more starters available.

III: Personal Applications

Spring Boot helps developers create new applications. In particular, it makes you to create standalone applications that work independently, without having to rely on an external web server, by embedding . During the startup process, you inject a web server such as Tomcat or Netty into your application so that you can run the application from any forum by simply clicking the "run" command, which results in from an external web server.

Spring Boot vs. Spring Framework

The great advantages of SpringBoot over Spring Framework alone are i.easy to use and ii. improve quickly. In theory, this is due to the great flexibility you find in working directly with the Spring Framework.

In practice, you need or want to use a unique configuration, using the Spring Booth , as it's worth the trade. You can still use the popular Spring Framework annotation system that allows you to easily inject additional dependencies (uncovered Spring Starters) into your app.

Also, you still get access to all the features of the Spring Framework, which include event management, authentication, data binding, genre conversion, and built-in protection and testing capabilities. Below, if the scope of your project is covered by even one Spring Starter, the Spring Boot can improve progress.

**REFERENCES**

1. Pluralsight.com
2. Morgan Stanley Material
3. Morgan Stanley Training Docs
4. Morgan Stanley Mentors
5. msinitializer/
6. cvaraman/
7. Bluestack
8. Google
9. Java documentation
10. Angular.io
11. YouTube
12. CVA Traders Group