Stay Ahead

BANNARI AMMAN

INSTITUTE OF TECHNOLOGY

An Autonomous Institution Affiliated to Anna University,
Approved by AICTE, Accredited by NAAC with 'A' Grade

INTERNSHIP REPORT

Submitted by

MANJU H (7376212AD160)

)

Internship
at

FORAGE - JPMorgan Chase & Co

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

BANNARI AMMAN INSTITUTE OF TECHNOLOGY (An

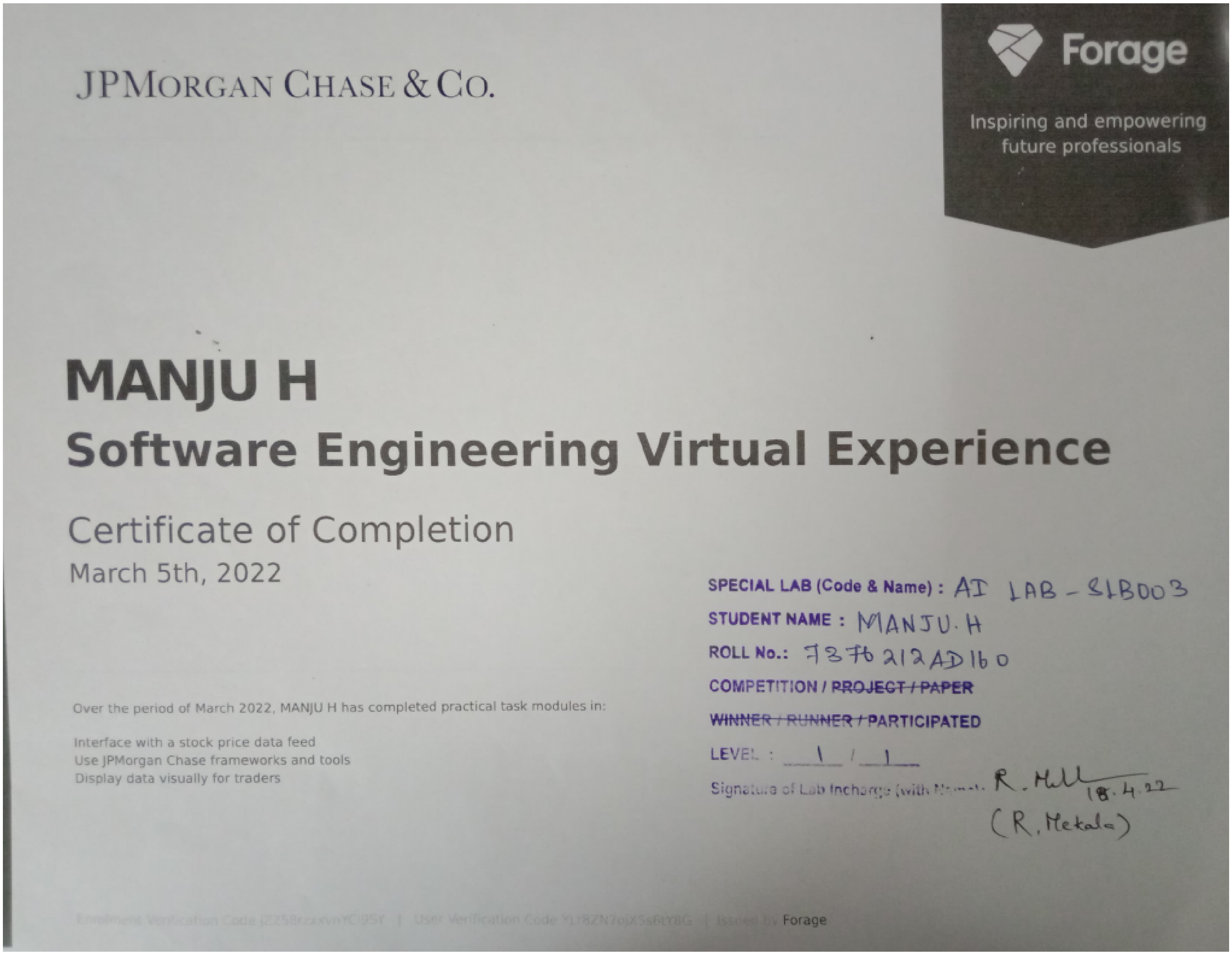
Autonomous Institution affiliated to Anna University, Chennai)

SATHYAMANGALAM – 638401

ANNA UNIVERSITY:CHENNAI

MARCH -2022

Internship certificate provided by the internship institution



JPMORGAN CHASE & CO.

Forage

Inspiring and empowering future professionals

MANJU H

Software Engineering Virtual Experience

Certificate of Completion

March 5th, 2022

Over the period of March 2022, MANJU H has completed practical task modules in:

Interface with a stock price data feed
Use JPMorgan Chase frameworks and tools
Display data visually for traders

SPECIAL LAB (Code & Name) : AI LAB - SLB003
STUDENT NAME : MANJU.H
ROLL No.: 7376212AD160
COMPETITION / ~~PROJECT / PAPER~~
~~WINNER / RUNNER /~~ PARTICIPATED
LEVEL : 1 / 1
Signature of Lab Incharge (with Name) : R. M 18.4.22
(R. Mekala)

Enrolment Verification Code jZ258rzxxvnYCl95Y | User Verification Code YLr8ZN7ojXSs6LY8G | Issued by Forage

ACKNOWLEDGEMENT

Firstly, I would like to thank **FORAGE - JPMorgan Chase & Co,** online platform for giving me the opportunity to do an internship within the organization.

(MANJU H-7376212AD160)

SUPERVISOR EVALUATION OF INTERN

Student Name: MANJU H Date: 15.02.2022 Work Supervisor:
Mr.S.KRISHNAKUMAR Title: MENTOR Company/Organization:
FORAGE - JPMorgan Chase & Co

Dates of Internship: From **15.02.2022** To **05.03.2022**

Please evaluate your intern by indicating the frequency with which you observed the following behaviors:

Parameters Needs improvement	Satisfactory Good
Behaviors	*
Performs in a dependable manner	*
Cooperates with co-workers and supervisors	*

Shows interest in work	*
Learns quickly	*
Shows initiative	*
Produces high quality work	*
Accepts responsibility	*

Accepts criticism	*
Demonstrates organizational skills	*
Uses technical knowledge and expertise	*
Shows good judgment	*
Demonstrates creativity/originality	*
Analyzes problems effectively	*
Is self-reliant	*
Communicates well	*
Writes effectively	*
Has a professional attitude	*

Gives a professional appearance	*
Is punctual	*
Uses time effectively	*

Overall performance of student
intern (circle one):

(Needs improvement/ Satisfactory/Good/ [✓ Excellent])
Additional comments, if any: NIL

Mr. KRISHNAKUMAR S

Mentor

ATTENDANCE SHEET

Name & Address of Organization: **FORAGE - JPMorgan Chase & Co**
Website Address :https://www.theforage.com/

Name of Student: **MANJU H**

Roll no. of the student: **7376212AD160**

Name of the Course: **SOFTWARE ENGINEERING VIRTUAL EXPERIENCE**

Year of Study: **I - YEAR**

Date of Commencement of Training: **15.02.2022**

Date of Completion of Training: **05.03.2022**

NO OF DAYS	DATE	ATTENDANCE
1	15.02.2022	SR
2	16.02.2022	SR
3	17.02.2022	SR
4	18.02.2022	SR
5	21.02.2022	SR
6	22.02.2022	SR
7	23.02.2022	SR
8	24.02.2022	SR
9	25.02.2022	SR
10	26.02.2022	SR
11	28.02.2022	SR
12	01.03.2022	SR
13	02.03.2022	SR
14	03.03.2022	SR
15	04.03.2022	SR

INTERNSHIP SYNOPSIS

INTERNSHIP OBJECTIVES

- **Knowledge and Understanding**

1. Basics of Software Engineering

2. Models of Software Engineering

3. Maintainability

- **Skills**

1. Teating
2. Debugging
3. Development of Software

Learning Activities:

This training helped me to understand the core concept of software engineering and the working and experience. I was able to learn to work with software applications.

On the Job:

During the training period, we are exposed to many new technologies of software engineering. Our trainer helped us to get the things very clearly.

Off the job:

We had a practical virtual experience by the end of the training. Also, we had an open discussion about that project through which we learnt communication skills and technical project presentation skills as well.

Job Description:

This virtual training provided the theoretical concepts of software engineering. I was assigned to manage people and work on projects with my fellow interns. At the end of the training, I was given a role to present my project ideas of software development.

Supervision:

My trainer and my mentor helped me in all aspects, cleared my confusions and doubts not only related to the training but also, they guided me personally about skills and attitude development.

STUDENT’S DAILY DIARY/ DAILY LOG

DAY-1			15.02.2022	
Time of arrival	5 PM	Time of Departure	6 PM	GOOD
Dept./Division	Software engineering virtual experience	Name of finished Product		

Name of HOD/ Supervisor With e-mail id Main points of the	**Mr.Krishnakumar s** krishnakumars@bitsathy.ac.in		

STUDENT’S DAILY DIARY/ DAILY LOG

DAY-2		DATE :16.02.2022
Time of arrival	5 PM Time of 6 PM Departure	GOOD

Dept/Devision	Software Name of engineering virtual finished experience Product	
Name of HOD/ Supervisor With e-mail id	**Mr.Krishnakumar s** krishnakumars@bitsathy.ac.in	
Main points of t	Basics of software engineering : development, maintenance	

STUDENT'S DAILY DIARY/ DAILY LOG

DAY-3		DATE: 17.02.2022	
Time of arrival	5 PM	Time of 6 PM Departure	GOOD

Dept./Division	Software engineering virtual experience	Name of finished Product
Name of HOD/ Supervisor With e-mail id		**Mr.Krishnakumar s** krishnakumars@bitsathy.ac.in
Main points		Software requirements : Portability, security, maintainability, reliability, scalability, performance, reusability, and flexibility.

STUDENT'S DAILY DIARY/ DAILY LOG

DAY-4		DATE: 18.02.2022	
Time of arrival	5 PM	Time of Departure 6 PM	GOOD

Dept./Division	Software engineering virtual experience	Name of finished Product
Name of HOD/ Supervisor With e-mail id		**Mr.Krishnakumar s** krishnakumars@bitsathy.ac.in
Main points of t		Software design : Correctness, completeness, efficiency, flexibility, consistently, maintainability.

STUDENT’S DAILY DIARY/ DAILY LOG

DAY-5		DATE: 21.02.2022	
Time of arrival	5 PM	Time of Departure 6 PM	GOOD

Dept./Division	Software engineering virtual experience	Name of finished Product
Name of HOD/ Supervisor With e-mail id Main points		**Mr.Krishnakumar s** krishnakumars@bitsathy.ac.in Software construction : Coding, debugging, detailed design, problem definition.

STUDENT’S DAILY DIARY/ DAILY LOG

DAY-6		DATE: 22.02.2022	
Time of arrival	5 PM	Time 6 PM of	GOOD

		Departure	
Dept./Divis io n	Software engineeri ng virtual experience	Name of finished Product	

Name of HOD/ Supervisor With e-mail id	**Mr.Krishnakumar s** krishnakumars@bitsathy.ac.i

STUDENT’S DAILY DIARY/ DAILY LOG

DAY-7		DATE : 23.02.2022	
Time of arrival	5 PM	of 6 PM Time Departure	GOOD

Dept./Divisio n	Software engineering virtual experience	Name of finished Product
Name of HOD/ Supervisor With e-mail id Main points		Mr.Krishnakumar s krishnakumars@bitsathy.ac.in Importance of software engineering.

STUDENT’S DAILY DIARY/ DAILY LOG

DAY-8		DATE: 24.02.2022	
Time of arrival	5 PM	Time of Departure 6 PM	GOOD

Dept./Divis io n	Software engineeri ng virtual experience	Name of finished Product
Name of HOD/ Supervisor With e-mail id Main points of t		**Mr.Krishnakumar s** krishnakumars@bitsathy.ac.in Black box testing : Unit,integration, system and acceptance.

STUDENT'S DAILY DIARY/ DAILY LOG

DAY-9		DATE: 25.02.2022	
Time of arrival	5 P.M	of 6PM	GOOD

		Time Departure	
Dept./Divisio n		of Name finished Product	
Name of HOD/ Supervisor With e-mail id Main points of t		**Mr.Krishnakumar s** krishnakumars@bitsathy.ac.in Functional testing .	

STUDENT'S DAILY DIARY/ DAILY LOG

DAY-10	DATE: 26.02.2022

Time of arrival	5 PM	of 6 PM Time Departure	GOOD
Dept./Division	Software engineering virtual experience	Name of finished Product	
Name of HOD/ Supervisor With e-mail id Main points of t		**Mr.Krishnakumar s** krishnakumars@bitsathy.ac.in Integration testing.	

STUDENT'S DAILY DIARY/ DAILY LOG

DAY-11	DATE: 28.02.2022

Time of arrival	5 PM	of 6 PM Time Departure	GOOD
Dept./Divis io n	Software engineeri ng virtual experience	Name of finished Product	
Name of HOD/ Supervisor With e-mail id Main points of t		**Mr.Krishnakumar s** krishnakumars@bitsathy.ac.in Waterfall model.	

STUDENT'S DAILY DIARY/ DAILY LOG

DAY-12		DATE: 01.03.2022	
Time of arrival	5 PM	Time of Departure 6 PM	GOOD
Dept./Divisio n	Software engineering virtual experience	Name of finished Product	
Name of HOD/ Supervisor With e-mail id Main points of t		**Mr.Krishnakumar s** krishnakumars@bitsathy.ac.in Agile model.	

STUDENT'S DAILY DIARY/ DAILY LOG

DAY-13		DATE: 02.03.2022	
Time of arrival	5 PM	Time 6 PM of Departure	GOOD
Dept./Divisio n	Software engineering virtual experience	Name of finished Product	
Name of HOD/ Supervisor With e-mail id Main points of t		**Mr.Krishnakumar s** krishnakumars@bitsathy.ac.in Iterative model.	

STUDENT'S DAILY DIARY/ DAILY LOG

DAY-14		DATE: 03.03.2022	
Time of arrival	5 PM	Time 6 PM of Departure	GOOD
Dept./Divisio n	Software engineering virtual experience	Name of finished Product	
Name of HOD/ Supervisor With e-mail id Main points of t		**Mr.Krishnakumar s** krishnakumars@bitsathy.ac.in Software maintenance.	

STUDENT'S DAILY DIARY/ DAILY LOG

DAY-15		DATE 04.03.2022	
Time of arrival	5 PM	Time of Departure 6 PM	GOOD
Dept./Divisio n	Software engineering virtual experience	Name of finished Product FINAL PROJECT	
Name of HOD/ Supervisor With e-mail id		**Mr.Krishnakumar s** krishnakumars@bitsathy.ac.in	
Main points of t		Final project.	

REPORT

TABLE OF CONTENTS

INTRODUCTION

LEARNING OBJECTIVES/INTERNSHIP OBJECTIVES

- Internships are generally thought of to be reserved for college students looking to gain experience in a particular field. However, a wide array of people can benefit from Training Internships in order to receive real world experience and develop their skills.

- An objective for this position should emphasize the skills you already possess in the area and your interest in learning more

- Internships are utilized in a number of different career fields, including architecture, engineering, healthcare, economics, advertising and many more.

- Some internships are used to allow individuals to perform scientific research while others are specifically designed to allow people to gain first-hand experience working.

- Utilizing internships is a great way to build your resume and develop skills that can be emphasized in your resume for future jobs. When you are applying for a Training Internship, make sure to highlight any special skills or talents that can make you stand apart from the rest of the applicants so that you have an improved chance of landing the position.

FORAGE

An E-learning platform, providing Courses, Training and Internship along with projects in many professional and technical courses. Forage provides various Training Skills to provide hands-on experience to individuals before starting their internship. Internship helps in providing practical knowledge of our subject. Forage provides courses cum internships like IoT, programming, business communication skills and many more through various interactive live Classes, project based activities and much more to get trained for an internship and job.

In order to provide you with all the necessary skills, this course also contains various job interview skills which will help you refine your personality. We are a technology company on a mission to equip students with relevant skills & practical exposure through internships and online training. We are Providing Free and Paid Online Trainings, Courses and Internships for college Students according to their field of passion. We also Organize Events, workshops for Corporates and bring Opportunities for Students. To give credibility to your learning, we provide you certification of Training, Letter of Recommendation & Internship opportunities at the end of course completion.

INTERNSHIP ACTIVITIES

SOFTWARE ENGINEERING VIRTUAL EXPERIENCE

DAY1: Introduction to the internship

On the first day, An internship is a period of work experience offered by an organization for a limited period of time. Once confined to medical graduates, internship is used for a wide range of placements in businesses, non-profit organizations and government agencies. They are typically undertaken by students and graduates looking to gain relevant skills and experience in a particular field. Employers benefit from these placements because they often recruit employees from their best interns, who have known capabilities, thus saving time and money in the long run. Internships are usually arranged by third-party organizations that recruit interns on behalf of industry groups. Rules vary from country to country about when interns should be regarded as employees. The system can be open to exploitation by unscrupulous employers.

Typically, an internship consists of an exchange of services for experience between the intern and the organization. Internships are used to determine whether the intern still has an interest in that field after the real-life experience. In addition, an internship can be used to build a professional network that can assist with letters of recommendation or lead to future employment opportunities. The benefit of bringing an intern into full time employment is that they are already familiar with the company, therefore needing little to no training. Internships provide current college students with the ability to participate in a field of their choice to receive hands-on learning about a particular future career, preparing them for full-time work following graduation.

DAY 2: BASIC INTRODUCTION OF SOFTWARE ENGINEERING

Software engineering is the systematic application of engineering approaches to the development of software.

A **software engineer** is a person who applies the principles of software engineering to design, develop, maintain, test, and evaluate computer software. The term programmer is sometimes used as a synonym, but may also lack connotations of engineering education orskills.

Engineering techniques are used to inform the software development process which involves the definition, implementation, assessment, measurement, management, change, and improvement of the software life cycle process itself. It heavily uses software configuration management which is about systematically controlling changes to the configuration, and maintaining the integrity and traceability of the configuration and code throughout the system life cycle. Modern processes use software versioning.

Software engineering is a detailed study of engineering to the design, development and maintenance of software. Software engineering was introduced to address the issues of low-quality software projects. Problems arise when a software generally exceeds timelines, budgets, and reduced levels of quality.

DAY 3: INTRODUCTION TO SOFTWARE REQUIREMENTS

Requirements engineering is about the elicitation, analysis, specification, and validation of requirements for software. Software requirements can be of three different types. There are functional requirements, non-functional requirements, and domain requirements. The operation of the software should be performed and the proper output should be expected for the user to use. Non- functional requirements deal with issues like portability, security, maintainability, reliability,

scalability, performance, reusability, and flexibility. They are classified into the following types: interference constraints, performance constraints (such as response time, security, storage space, etc.), operating constraints, life cycle constraints (maintainability, portability, etc.), and economic constraints. Knowledge of how the system or software works is needed when it comes to specifying non-functional requirements. Domain requirements have to do with the characteristic of a certain category or domain of projects

DAY 4: OBJECTIVES OF SOFTWARE DESIGN

Software design is a mechanism to transform user requirements into some suitable form, which helps the programmer in software coding and implementation. It deals with representing the client's requirement, as described in the SRS (Software Requirement Specification) document, into a form, i.e., easily implementable using programming language.

The software design phase is the first step in SDLC (Software Design Life Cycle), which moves the concentration from the problem domain to the solution domain. In software design, we consider the system to be a set of components or modules with clearly defined behaviors & boundaries.

1. **Correctness:**Software design should be correct as per requirement.
2. **Completeness:**The design should have all components like data structures, modules, and external

interfaces, etc.

3. **Efficiency:**Resources should be used efficiently by the program.
4. **Flexibility:**Able to modify on changing needs.
5. **Consistency:**There should not be any inconsistency in the design.
6. **Maintainability:** The design should be so simple so that it can be easily maintainable by other designers.

DAY 5: SOFTWARE CONSTRUCTION

The collection of activities that actually produce the working software.

The diagram here shows several component activities that should be familiar to you from our earlier discussion of process models. The size of each cloud is intended to suggest the relative amount of time and effort spent in that activity. The position of the cloud, relative to the yellow oval that defines "software construction", indicates whether that activity is central or peripheral to software construction, and whether it is entirely a software construction activity (e.g., coding and debugging), partially software construction (e.g., detailed design), or not part of software construction at all (e.g., problem definition).

- Coding and debugging at the center, plus supporting activities
- Excludes management, reqts. analysis, architectural design, UI design, acceptance testing, maintenance
- "the only activity that's guaranteed to be done" (McConnell)
- the fun part

DAY 6: INTRODUCTION TO SOFTWARE TESTING

Software Testing is a method to check whether the actual software product matches expected requirements and to ensure that the software product is Defect free. It involves execution of software/system components using manual or automated tools to evaluate one or more properties of interest. The purpose of software testing is to identify errors, gaps or missing requirements in contrast to actual requirements.

Some prefer saying Software testing definition as a White Box and Black Box Testing. In simple terms, Software Testing means the Verification of Application Under Test (AUT). This Software Testing course introduces testing software to the audience and justifies the importance of software testing. Software testing is an empirical, technical investigation conducted to provide stakeholders with information about the quality of the product or service under test, with different approaches such as unit testing and integration testing. It is one aspect of software quality. As a separate phase in software development, it is typically performed by quality assurance staff or a developer other than the one who wrote the code.

DAY 7: IMPORTANCE OF SOFTWARE ENGINEERING

Testing is important because software bugs could be expensive or even dangerous. Software bugs can potentially cause monetary and human loss, and history is full of such examples. ● In April 2015, the Bloomberg terminal in London crashed due to a software glitch affecting more than 300,000 traders on financial markets. It forced the government to postpone a 3bn pound debt sale.

● Nissan cars recalled over 1 million cars from the market due to software failure in the airbag sensory detectors. There have been two reported accidents due to this software failure. ● Starbucks was forced to close about 60 percent of stores in the U.S and Canada due to software failure in its POS system. At one point, the store served coffee for free as they were unable to process the transaction.

- Some of Amazon's third-party retailers saw their product price reduced to 1p due to a software glitch. They were left with heavy losses.
- Vulnerability in Windows 10. This bug enables users to escape from security sandboxes through a flaw in the win32k system.
- In 2015 fighter plane F-35 fell victim to a software bug, making it unable to detect targets correctly.
- China Airlines Airbus A300 crashed due to a software bug on April 26, 1994, killing 264 innocents live
- In 1985, Canada's Therac-25 radiation therapy machine malfunctioned due to software bug and delivered lethal radiation doses to patients, leaving 3 people dead and critically injuring 3 others.
- In April of 1999, a software bug caused the failure of a $1.2 billion military satellite launch, the costliest accident in history
- In May of 1996, a software bug caused the bank accounts of 823 customers of a major U.S. bank to be credited with 920 million US dollars.

DAY 8: INTRODUCTION TO BLACK BOX TESTING

In this testing, the test engineer will analyze the software against requirements, identify the defects or bugs, and sends it back to the development team.Then, the developers will fix those defects, do one round of White box testing, and send it to the testing team. Here, fixing the bugs means the defect is resolved, and the particular feature is working according to the given requirement.The main objective of implementing the black box testing is to specify the business needs or the customer's requirements. In other words, we can say that black box testing is a process of checking the functionality of an application as per the customer requirement. The source code is not visible in this testing; that's why it is known as black-box testing. Black box testing involves testing a system with no prior knowledge of its internal workings. A tester provides an input, and observes the output generated by the system under test.This method is named so because the software program, in the eyes of the tester, is like a black box; inside which one cannot see.

DAY 9: INTRODUCTION TO FUNCTIONAL TESTING

The test engineer will check all the components systematically against requirement specifications known as functional testing. Functional testing is also known as Component testing. In functional testing, all the components are tested by giving the value, defining the output, and validating the actual output with the expected value. Functional testing is a part of black-box testing as its emphases on application requirements rather than actual code. The test engineer has to test only the program instead of the system. Functional testing is the process through which QAs determine if a piece of software is acting in accordance with predetermined requirements. It uses black-box testing techniques, in which the tester has no knowledge of the internal system logic. It is a type of software testing which is used to verify the functionality of the software application, whether the function is working according to the requirement specification. In functional testing, each function is tested by giving the value, determining the output, and verifying the actual output with the expected value. Functional testing performed as black-box testing which is presented to confirm that the functionality of an application or system behaves as we are expecting. It is done to verify the functionality of the application.Functional testing also called as black-box testing, because it focuses on application specification rather than actual code. Tester has to test only the program rather than the system.

DAY 10: INTRODUCTION TO INTEGRATION TESTING

Once we are successfully implementing the unit testing, we will go integration testing. It is the second level of functional testing, where we test the data flow between dependent modules or interface between two features are called integration testing.Integration testing is the second level of the software testing process after unit testing. In this testing, units or individual components of the software are tested in a group. The focus of the integration testing level is to expose defects at the time of interaction between integrated components or units.Unit testing uses modules for testing purpose, and these modules are combined and tested in integration testing. The Software is developed with a number of software modules that are coded by different coders or programmers. The goal of integration testing is to check the correctness of communication among all the modules.

DAY 11: WATERFALL MODEL

Winston Royce introduced the Waterfall Model in 1970.This model has five phases: Requirements analysis and specification, design, implementation, and unit testing, integration and system testing, and operation and maintenance. The steps always follow in this order and do not overlap. The developer must complete every phase before the next phase begins. This model is named "Waterfall Model", because its diagrammatic representation resembles a cascade of waterfalls.

1. Requirements analysis and specification phase: The aim of this phase is to understand the exact requirements of the customer and to document them properly. Both the customer and the software developer work together so as to document all the functions, performance, and interfacing requirements of the software. It describes the "what" of the system to be produced and not "how."In this phase, a large document called Software Requirement Specification (SRS) document is created which contains a detailed description of what the system will do in the common language**.**

2. Design Phase: This phase aims to transform the requirements gathered in the SRS into a suitable form which permits further coding in a programming language. It defines the overall software architecture together with high level and detailed design. All this work is documented as a Software Design Document (SDD).

3. Implementation and unit testing: During this phase, design is implemented. If the SSD is complete, the implementation or coding phase proceeds smoothly, because all the information needed by software developers is contained in the SDD.During testing, the code is thoroughly examined and modified. Small modules are tested in isolation initially. After that these modules are tested by writing some overhead code to check the interaction between these modules and the flow of intermediate output.

4. Integration and System Testing: This phase is highly crucial as the quality of the end product is determined by the effectiveness of the testing carried out. The better output will lead to satisfied customers, lower maintenance costs, and accurate results. Unit testing determines the efficiency of individual modules. However, in this phase, the modules are tested for their interactions with each other and with the system.

5. Operation and maintenance phase: Maintenance is the task performed by every user once the software has been delivered to the customer, installed, and operational.

DAY 12: INTRODUCTION TO AGILE MODEL

The meaning of Agile is swift or versatile."Agile process model" refers to a software development approach based on iterative development. Agile methods break tasks into smaller iterations, or parts do not directly involve long term planning. The project scope and requirements are laid down at the beginning of the development process. Plans regarding the number of iterations, the duration and the scope of each iteration are clearly defined in advance.

Each iteration is considered as a short time "frame" in the Agile process model, which typically lasts from one to four weeks. The division of the entire project into smaller parts helps to minimize the project risk and to reduce the overall project delivery time requirements. Each iteration involves a team working through a full software development life cycle including planning, requirements analysis, design, coding, and testing before a working product is demonstrated to the client.

Agile Testing Methods:

- Scrum
- Crystal
- Dynamic Software DevelopmentMethod(DSDM)
- Feature Driven Development(FDD)
- Lean Software Development
- eXtreme Programming(XP)

DAY 13: INTRODUCTION TO ITERATIVE MODEL

In this Model, you can start with some of the software specifications and develop the first version of the software. After the first version if there is a need to change the software, then a new version of the software is created with a new iteration. Every release of the Iterative Model finishes in an exact and fixed period that is called iteration. The

Iterative Model allows accessing earlier phases, in which the variations are made respectively. The final output of the project was renewed at the end of the Software Development Life Cycle (SDLC) process.

Requirement gathering & analysis: In this phase, requirements are gathered from customers and checked by an analyst whether requirements will be fulfilled or not. Analyst checks that need will achieve within budget

or not. After all of this, the software team skips to the next phase.

2. Design: In the design phase, the team designs the software by the different diagrams like Data Flow diagram, activity diagram, class diagram, state transition diagram, etc.

3. Implementation: In the implementation, requirements are written in the coding language and transformed into computer programmes which are called Software.

4. Testing: After completing the coding phase, software testing starts using different test methods. There are many test methods, but the most common are white box, black box, and gray box test methods.

5. Deployment: After completing all the phases, software is deployed to its work environment. **6.**

Review: In this phase, after the product deployment, review phase is performed to check the behavior

and validity of the developed product. And if there are any errors found then the process starts again from the requirement gathering.

7. Maintenance: In the maintenance phase, after deployment of the software in the working environment there may be some bugs, some errors or new updates are required. Maintenance involves debugging and new addition options.

DAY 14: SOFTWARE MAINTENANCE

Software maintenance is a part of the Software Development Life Cycle. Its primary goal is to modify and update software applications after delivery to correct errors and to improve performance. Software is a model of the real world. When the real world changes, the software requires alteration wherever possible.

Software Maintenance is an inclusive activity that includes error corrections, enhancement of capabilities, deletion of obsolete capabilities, and optimization. Software maintenance refers to the activities required to provide cost-effective support after shipping the software product. Software maintenance is modifying and updating software applications after distribution to correct faults and to improve its performance.

Software has a lot to do with the real world and when the real world changes, software maintenance is required. Software maintenance includes: error correction, optimization, deletion of unused and discarded features, and enhancement of features that already exist. Usually, maintenance takes up about 40% to 80% of the project cost therefore, focusing on maintenance keeps the costs down.

DAY 15: FINAL PROJECT SUBMISSION

On the final day assessment was conducted and certificates were issued to the qualified students. This was a great internship and the work experience was great even on the online platform.

PERFORMANCE FOR EVALUATION OF INTERNSHIP BY INSTITUTE

DEPARTMENT OF TRAINING AND PLACEMENT

Phone.No : 8248959097 Fax: **NIL**

Email:shona.ad21@bitsathy.ac.in

Evaluation (I)

1. Name of Student: **MANJU H** Mob. No: **8248959097**

2. College Roll No.: **7376212AD160**

3. Branch/Semester: **AI & DS / SEM-I**

Period of Training: **15 DAYS**

4. Home Address with contact No:

4/218,
KAGGUCHI VILLAGE ,
KAGGUCHI POST,
KOTAGIRI,
THE NILGIRIS- 643214

Address of Training Site: **ONLINE**

5. Address of Training Providing Agency: **FORAGE - JPMorgan Chase & Co**

6. Name/Designation of Training In- charge: **Self**

7. Type of Work: **INDUSTRIAL TRAINING - VIRTUAL EXPERIENCE 8.** Date of Evaluation: 19.02.2022

a) Attendance: _ (Satisfactory/ Good/ **Excellent**)

b) Practical Work: (Satisfactory/ Good/ **Excellent)**

c) Faculty's Evaluation: _ (Satisfactory/ Good/ **Excellent**)

d) Evaluation of Industry: _(Satisfactory/ Good/ **Excellent**)

9. Overall grade: (Satisfactory/ Good/ **Excellent**)

Mr.KRISHNAKUMAR. S	Mrs. GAYATHRI. J
Signature of Faculty advisor	**Signature of the HOD**