

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/350896282>

6 Month Internship as a Trainee Software Engineer

Technical Report · April 2021

CITATIONS

0

READS

576

1 author:



Asith Ishantha
Future University Hakodate

5 PUBLICATIONS 0 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



MALL CUSTOMER SEGMENTATION USING CLUSTERING ALGORITHM [View project](#)



Impact of Supermarkets on Traditional Markets and Retailers in Sri Lanka's Western Province [View project](#)

National Apprentice & Industrial

Training Authority

Report on Industrial Training

At

hSenid Software International

Colombo 07



Lanka Nippon Biztech Institute

Maharagama

Name : M.A.Ishantha

Student Number : NVQ/LNBTI/00005/2020

Course : Higher Diploma in Information
Technology

Field : Information Technology

Training Period : 20.07.2020 – 20.01.2021

Acknowledgement

First of all, I would like to thank the Management and Academic staff of Lanka Nippon Biztech Institute and Industrial Training Authority (NAITA) for providing me with an Industrial experience in addition to our Higher Diploma.

I am happy to record my thanks to the HR department of hSenid Software International (PVT) Ltd, for organizing online introduction sessions, which were important to get an idea about company staff, common policies and working environment of the company.

I owe a debt of gratitude to Mr. Vimukthi Dissanayake, Manager Engineering and Tech Lead, Mr. Danasiri Wijedasa, Project Manager. Mr.Tharuke Jayawardhane, Associate Software Engineer and Mr Menuka Perera, Associate Software Engineer, to continuous supervision of the team of interns throughout the internship period.

I should especially be grateful to Tech Lead, Mr. Vimukthi Dissanayake for giving me an opportunity to have broad practical knowledge about new technologies, mainly in Java and giving responsibilities in hSenid company internal projects such as Applova Japan, JobsDB and Hutch Danumai Miliyanayai app, Kitcut Japan and Furthermore, I am happy to mention Mr.Tharuke Jayawardhane ,Associate Software Engineer and Mr.Menuka Perera, Associate Software Engineer for giving me wide knowledge and precious support.

Finally, I would like to thank all the members of the team and everyone who collaborated with me during the industrial training period in hSenid Software International.

M.A.Ishantha

Department of Software Engineering

Lanka Nippon BizTech Institute

Preface

This Industrial Training Report is based on, which I performed after completion of 27 weeks of Industrial training at hSenid Software International (Pvt)Ltd. Industrial Training Completion of 27 weeks is compulsory for the award of the NVQ level 6 diploma from the Lanka Nippon BizTech Institute, Sri Lanka

Industrial Training program was held on 6th semester of the Software engineering diploma program curriculum and Industrial training is carried out by the National Apprentice and Industrial Training Authority (NAITA) in collaboration with the LNBTI. This report contained experiences and knowledge which I gathered during my training period from 20/07/2020 to 20/01/2021.

Mainly report has three Chapters. Chapter 1 includes information about training organization, project description, hierarchy of the organization, safety of the project and welfare activities. Chapter 2 mainly described my experiences which are learned during my training and comparison with structural knowledge which I gained during academic. Chapter 2 is divided into sub topics according to which I performed and observed construction activities on the site.

Finally, Chapter 3 is about the conclusion of the training, which includes my development, suggestions and activities which I couldn't perform.

Contents

Acknowledgement	2
Preface.....	3
List of figures	5
Chapter 1 - Introduction about training Institute	12
1.1 About Training organization	12
1.1.1 Company overview	12
1.1.2 Nature of business.....	15
1.2 Co – operator plan.....	21
1.2.1 Vision.....	21
1.2.2 Mission.....	21
1.2.3 Milestones	21
1.2.4 Values	22
1.2.5 Number of employees (hSenid Software Lanka).....	22
1.3 Management practices.....	22
1.3.1 Leave information.....	22
1.3.2 EPF and ETF.....	22
1.3.3 Company offers.....	23
1.3.4 Recruitment Procedure.....	23
1.3.5 Security policy	23
1.3.6 Compensation	24
Chapter 2 - Training Experience.....	24
2.1 Introduction	24
2.2 My Training Experience and Actions Performed	24
2.3 Tools, Technologies and Applications	119
2.4 Work and Developments during the internal training period.....	123
2.41 Applova Testing	123
2.42 “HUTCH Danumai Miliyanayai”(Sinhala Tamil Unicode)	140
2.43 BeyondM messaging platform testing	141
2.44 Web Scraping Tabelogu Japan and UberEats Japan	146
2.45 JobsDB testing.....	158
2.46 JobsDB development.....	162
2.47 Kitcut testing.....	176
2.48 Final day of the Internship.....	187

Chapter 3 - Conclusion	187
------------------------------	-----

List of figures

Figure 1:Organization hierarchy	13
Figure 2: Organization structure	13
Figure 3:Dinesh B. Saparamadu	14
Figure 4:Sampath Jayasundara	14
Figure 5:Sujee Saparamadu	14
Figure 6:Cpass	15
Figure 7:Cube.....	16
Figure 8:Telco-Signaling Platforms.....	16
Figure 9:Bank digitalization platform.....	17
Figure 10:Peoples HR	17
Figure 11:Tracking solution.....	18
Figure 12:Juraa.....	18
Figure 13:HRO.....	18
Figure 14:Revenue management.....	19
Figure 15:Hrm enterprise	19
Figure 16:Software Development	19
Figure 17:Mobile App Development	20
Figure 18:Consulting Services	20
Figure 19:Co – operator plan	21
Figure 20:curl	25
Figure 21:Install Curl1	25
Figure 22:Install Curl2	26
Figure 23:curl -o abc.txt https://www.facebook.com	26
Figure 24:Figure 23:curl -o abc.txt https://www.facebook.com2.....	27
Figure 25:Get the response from GET request method	27
Figure 26:Get the response and headers using GET request method	28
Figure 27:POST method	28
Figure 28:PUT method	28
Figure 29:i flag.....	29
Figure 30:Delete method.....	29
Figure 31:training plan.....	30
Figure 32:Directory Structure	33
Figure 33:lowercase L and lowercase S.....	34
Figure 34:To list all files in your home directory	34
Figure 35:Mkdir directory.....	35
Figure 36:copy files from one directory to another directory	35
Figure 37:Move files from one directory to another.....	36
Figure 38:Cat command.....	36
Figure 39:grep command	37

Figure 40:-i use to ignore case sensitive	37
Figure 41:Cat redirection	37
Figure 42:Redirecting the output	38
Figure 43:Using >> symbol, we can do more changes to file.	38
Figure 44:Users can join two files using cat command	39
Figure 45:Sort	39
Figure 46:Using < symbol, we can set to get input from the file.	39
Figure 47:who	40
Figure 48:Who sort	40
Figure 49:Cat >> list	40
Figure 50:Ls list* - filter words starts with list word.....	41
Figure 51:Ls ?ist -? Represent exactly one character	41
Figure 52:apropos keyword	41
Figure 53:ls -l.....	42
Figure 54:chmod oug+x people	42
Figure 55:ps	43
Figure 56:ls	43
Figure 57:!sudo	44
Figure 58: cat>asith.txt	45
Figure 59:#! usr/bin/bash	46
Figure 60: ./bashscript.sh	46
Figure 61: vi bashscript.sh	46
Figure 62:./bashscript.sh	47
Figure 63: bashscript program	47
Figure 64:If Statements bashscript.....	47
Figure 65:If Statements bashscript.....	48
Figure 66: #! usr/bin/bash	48
Figure 67:who grep a	48
Figure 68:bashscript function	49
Figure 69:bashscript function function2	49
Figure 70:SED	50
Figure 71:SED1	50
Figure 72:mysql new group	52
Figure 73:check mysql connection	53
Figure 74:mysql create new user	53
Figure 75:gitlab github.....	54
Figure 76:creating a git hub	57
Figure 77:pushed to a remote.....	57
Figure 78:create a README file	57
Figure 79:git add" command.....	58
Figure 80>Create a repository on GitHub.....	58
Figure 81:to push the local repository contents into the remote host repository	58
Figure 82:git push origin master	59
Figure 83:updated git repository.....	59
Figure 84:by git commit -m 'your message',.....	59

Figure 85:git remote add origin 'your_URL_name'	60
Figure 86:git push -u origin <master or branch name>	60
Figure 87:View your files in repository hosted on git lab.	60
Figure 88:Using Command line to PULL to Git lab	61
Figure 89:I will check out my branch and pull updated code.....	61
Figure 90:Git tag	61
Figure 91:Inheritance is one of the key features of OOP.....	62
Figure 92:Inheritance java program.....	63
Figure 93:Java Inheritance main program	64
Figure 94:Java inheritance output.....	64
Figure 95:Java method overriding.....	64
Figure 96:Java overriding program.....	65
Figure 97:Overriding main class.....	66
Figure 98:Output.....	66
Figure 99:overriding main class.....	66
Figure 100:iveriridn gsuperclass subclass	67
Figure 101:suoerclass subclass output	67
Figure 102:lambda expression	68
Figure 103:lambda functional interface	68
Figure 104:define lambda expressions in Java	70
Figure 105:writing lambda expression in another way.....	70
Figure 106:A body with a single expression.....	71
Figure 107:a block body	71
Figure 108:Single abstract method	72
Figure 109:Lambda expression abstract method	73
Figure 110:lambda expression output.....	73
Figure 111:lambda expression abstract program	74
Figure 112:Lambda Expressions with parameters	74
Figure 113:lambda implementation	75
Figure 114: Lambda expression output.....	75
Figure 115:Java streams.....	75
Figure 116:Array List Lambda expression	76
Figure 117:Output lambda expression	76
Figure 118:lambda expression filter	77
Figure 119:IS vs NIO.....	77
Figure 120:Stream Oriented.....	78
Figure 121:Buffer Oriented.....	78
Figure 122:Set breakpoints	79
Figure 123:Program Arguments	80
Figure 124:arguments apply to the program.....	80
Figure 125::arguments apply to the program2.....	81
Figure 126::arguments apply to the program3.....	81
Figure 127::arguments apply to the program4.....	81
Figure 128:Feature of Servlet	83
Figure 129:Hierarchy of packages	84

Figure 130:Signature of service() method:	85
Figure 131:HTTP Servlet.....	85
Figure 132:the JSP life cycle	88
Figure 133:Example_jsp.java	89
Figure 134:Example_jsp.java output	89
Figure 135:complex operations JSP.....	91
Figure 136:JSP directives	92
Figure 137:JSP include Directive	92
Figure 138:Servlet bank app login.....	93
Figure 139:bank app login info.....	93
Figure 140:bank app form validation.....	94
Figure 141:account number must be filled out".	94
Figure 142: bank amount must be filled out".	95
Figure 143:withdraw money and deposit money	95
Figure 144:withdraw money enter amount	96
Figure 145:successfully withdrawal	96
Figure 146:bank app deposit money	97
Figure 147:successfully deposited	97
Figure 148:Insufficient balance	98
Figure 149:mvn -version	99
Figure 150:maven pom	100
Figure 151:Deposit servlet class- (test class / test method)	101
Figure 152:withdraw servlet class- (test class / test method)	102
Figure 153:select the necessary module	106
Figure 154:JavaFX SDK 14 together with OpenJDK 14, add one more VM option: --add-exports javafx.graphics/com.sun.javafx.sg.prism=ALL-UNNAMED	107
Figure 155:Run the application.....	108
Figure 156:Dispatcher Servlet	109
Figure 157:Hibernate bankapp.....	111
Figure 158:Hibernate login	111
Figure 159:Hibernate registration	112
Figure 160:Thanks for registering	112
Figure 161:Login was unsuccessful.....	113
Figure 162:homepage.....	113
Figure 163:After withdraw money UI interface	113
Figure 164:REST controller spring boot.....	114
Figure 165:REST controller spring boot2.....	114
Figure 166:Hire purchase module.....	115
Figure 167:Hire purchase module payment schedule	116
Figure 168:Spring boot backend rest API logic	117
Figure 169:Spring boot backend rest API logic2.....	118
Figure 170:Spring boot backend rest API logic3.....	118
Figure 171:Angular frontend	119
Figure 172:created a category called 屋台食べ物。	123
Figure 173:pressed the 編集	124

Figure 174:fill information regarding category	124
Figure 175:商品に関する画像).....	125
Figure 176:press, 更新(update),	125
Figure 177:press 無効化 feature provides unavailability for the category.....	126
Figure 178:we press the same button again, we must be able to see the availability of categories	126
Figure 179:confirmation of the re availability.....	126
Figure 180:Confirmation of re availabillity of item	127
Figure 181:confirmation of updates.....	127
Figure 182:1. Commodity display order change 商品の表示順の変更	127
Figure 183:created a sub category called お好み焼き here and 追加(add) press to add it into sub topic.....	128
Figure 184:After 更新 press this. We will be able to see it as a subcategory of 屋台食べ物 category.....	128
Figure 185:Item section	129
Figure 186:Available items.....	129
Figure 187:Takoyaki japanese food.....	129
Figure 188:change the description	130
Figure 189:confirmation of change the description	130
Figure 190:make carbon copy of the item	131
Figure 191:two items	131
Figure 192:オプションのグループ管理(option group management)	132
Figure 193:finished selection	132
Figure 194:Add on option for food.....	133
Figure 195:ask to add more images	133
Figure 196:add images location	133
Figure 197:adding new items location	134
Figure 198: item search functionality	135
Figure 199:search curry item	135
Figure 200:Display all items	136
Figure 201:customer side main interface	136
Figure 202:add items without limit.....	137
Figure 203:add on salt. sugar for each food item.....	138
Figure 204:delete all items in the cart	138
Figure 205:cart	139
Figure 206:Cart total price	139
Figure 207:受け取り方法の選択” (accept payment method selection)	140
Figure 208:HUTCH Danumai Miliyanayai	140
Figure 209:Danumai Miliyanayai excel sheet	141
Figure 210:Beyondm login interface	141
Figure 211:BeyondM support interface	142
Figure 212:Add mobile number of subscribers here.....	142
Figure 213:BeyondM campaign report	143
Figure 214:Credit Usage Report.....	143

Figure 215:Subscriber information	144
Figure 216:send sms interface	144
Figure 217:schedule the campaigns	145
Figure 218:Tabelogu website.....	146
Figure 219:Uber Eats Japan homepage.....	146
Figure 220:JobsDB homepage	158
Figure 221:cv templates	159
Figure 222:cv generation UI	162
Figure 223:Available templates	163
Figure 224:cv template without subject.....	164
Figure 225:postman request.....	168
Figure 226:Deleting existing templates	169
Figure 227:cv template with subject	170
Figure 228: cv blue without subject full cv	171
Figure 229:Peradeniya template	172
Figure 230:Peradeniya blue template.....	173
Figure 231:Yellow template	174
Figure 232:Yellow template	175
Figure 233:Kitcut Japan	176
Figure 234:Kitcut login.....	177
Figure 235:Kitcut registration.....	178
Figure 236:confirmation email sent message	178
Figure 237:Confirmation email.....	179
Figure 238:Kitcut Homepage.....	179
Figure 239:Forget password interface – email type.....	180
Figure 240:Reset password link sent to email box email notice.....	181
Figure 241:Password reset email	181
Figure 242:Enter new password and re-confirm password.....	182
Figure 243:username box error message- textbox validation	182
Figure 244:Empty email box validation test.....	183
Figure 245:Empty password box validation test.....	183
Figure 246>Error message called- “username is already used”	184

List of Abbreviations

HSL	hSenid Software Lanka
CEO	Chief Executive Officer
CTO	Chief Technology Officer
OS	Operating System
DB	Database
IDE	Integrated Development Environment
SQL	Structured Query Language
JSON	JavaScript ObjectNotation
API	Application Programming Interface
REST	Representational State Transfer
HTTP	Hypertext Transfer Protocol
URL	Uniform Resource Locator
BA	Business Analyst
PM	Project Manager
IOS	Internetwork Operating system
CSS	Cascading Style Sheets
UI/UX	User Interface / User Experience
GPS	Global Positioning System
HR	Human Relation

Chapter 1 - Introduction about training Institute

1.1 About Training organization

1.1.1 Company overview

In 1997, hSenid Software International was founded as a software company. hSenid is an application and service provider for the telecom, financial and enterprise markets. They specialized in human resource applications and mobile applications and wireless markets. hSenid is also widely recognized for its reliable offshore/outsourcing capabilities.

Over time, hSenid diversified into mobile solutions which is now a multinational Telco-Mobile Software Solutions provider catering primarily to the telecom and financial markets across the globe.

Then they moved into outsourcing business too. hSenid Outsourcing manages and delivers complex global outsourcing projects that include product conceptualization till end product delivery & customer support, which has made us a thought leader in outsourcing business.

More recently they launched a plug and play outfit called LOFT1024, an incubator which gives start-ups and other entrepreneurs the opportunity to work out of an easily accessible location in Colombo.

With Sri Lanka being recognized as a vital tourist destination, they unveiled Mount Havana, a holiday home located in the panoramic hills of Gampola. Managed by hSenid, Mount Havana has villa-type luxury suites and practices and supports sustainable living and eco-friendly methods, which are woven into its ethos of community mindfulness.

Today, hSenid is internationally recognized, having won numerous worldwide accolades and peer acknowledgments. They also operate from the USA, Australia, Singapore, Kenya and India. Currently they serve over 800,000 users and over 1000 clients globally across 18 industries in 30 countries.

hSenid Software International - Organization Structure

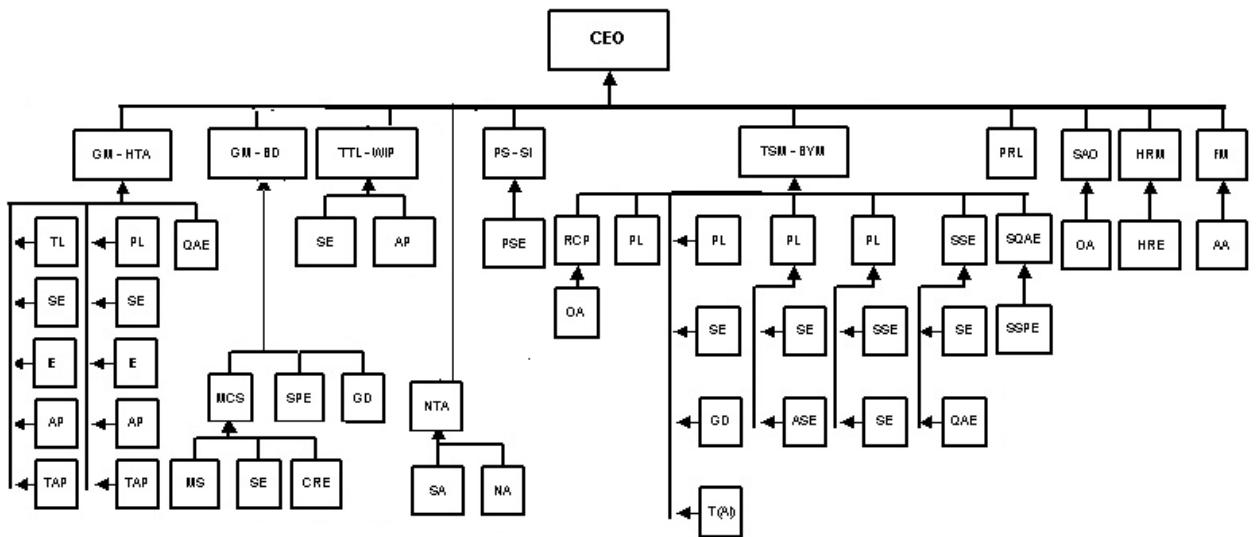


Figure 1: Organization hierarchy

CEO	- Chief Executive Officer	FM	- Finanace Manager
GM (HTA)	- General Manager Projects & Product Development	AA	- Account Assistant
GM (BD)	- General Manager Business Development	RCP	- Receptionist
TTL	- Technology Team Leader	OA	- Office Assistant
TSM	- Technology Service Manager	SA	- System Administrator
TL	- Team Leader	NTA	- Network Administrator
PL	- Project Leader	NA	- Network Assistant
PS	- Product Specialist	BA	- Business Analyst
SSE	- Senior S/W Engineer	PA	- Analyst Programmer
MCS	- Marketing & Communication Specialist	TAP	- Trainee Analyst Programmer
SE	- Sales Engineer	IE	- Implementation Engineer
SWE	- S/W Engineer	PRL	- Process Lead
SPE	- Support Engineer	CRE	- Customer Relationship Executive
TW	- Technical Writer	SAO	- Senior Administration Officer
GD	- Graphics Designer	MS	- Marketing Specialist
HRE	- Human Resource Executive	SQAE	- Senior QA Engineer
HRM	- Human Resource Manager	SSPE	- S/W Support Engineer
AP	- Analyst Programmer	DRA	- Documentation Research Assistant
PSE	- Pre Sales Engineer	ASE	- Associate S/W Engineer

Figure 2: Organization structure



*Figure 4:Dinesh B.
Saparamadu*

Dinesh B. Saparamadu
Chairman hSenid
Group of Companies



*Figure 3:Sampath
Jayasundara*

Sampath Jayasundara
Chief Executive Officer
hSenid Business Solutions



Figure 5:Sujee Saparamadu

Sujee Saparamadu
Director

1.1.2 Nature of business

1.1.2.1 Products

- CPaaS
 - hSenid Mobile's CPaaS (Communication Platform as a Service) is the perfect solution for Telcos' to expose the wealth of communication assets they have, making different communication mediums available on hassle free pay-as-you-go charging models, to anyone with a need for enabling communication to advance digital transformation and customer engagement.



Figure 6: Cpass

- CUBE
 - CUBE means Customized User Bound Experience which is a turnkey business-to-consumer fidelity program management engine designed as an intelligent analytical engine to analyze the customer behaviors along with segmentation and profiling. CUBE analyzes customers and their behavioral patterns individually to perform a precise segmentation, resulting in insightful profiles that could help Telcos to reduce churn, increase ARPU and optimize internal operational efficiencies.



Figure 7:Cube

- Telco-Signaling Platform
 - Telco-Signaling Platforms enable Telcos to establish relations between businesses and consumers through various Telco services. Use of an in-house built SIGTRAN stack lowers the Total Cost of Ownership (TCO) and eliminates the need to purchase an expensive 3rd party SIGTRAN stack or SS7 card.



Figure 8:Telco-Signaling Platforms

\

- Bank Digitalization Platform

A digital platform to bridge the gap between the customer expectation, cost management and internal process management, as it connects legacy systems, Omni channel distribution and internal workflow into one platform.



Figure 9:Bank digitalization platform



Figure 10:Peoples HR

A cloud-based HR solution that will engage and empower employees to build competitive organizations that are constantly achieving desired goals.



Figure 11: Tracking solution

Tracking systems designed and developed to secure and manage assets of companies, whether it be employees, places or objects.



Figure 12: Juraa

An Enterprise Social Network (ESN) that comes with next generation features which lets employees connect, collaborate and engage with their peers.



Figure 13: HRO

A HR process management entity that streamlines deliverables, expedites workflows and propels growth, enabling higher concentration to core business functions without distraction.



Figure 14:Revenue management

A system engineered and developed to be utilized by a local government body in a country, with an aim of increasing efficiency and effectiveness in tax and other revenue collections.



Figure 15:Hrm enterprise

A comprehensive Human Resources Management and Information system that caters to a wider spectrum of HR functions to improve the HR business processes of organisations.

- Software Development

With strategic alliances with technology providers and ISO and CMMI certified processes, we provide application outsourcing facilities that will align IT to support business functionalities that will provide a quality solution to effectively achieve business objectives.



Figure 16:Software Development

- Mobile App Development

Proposing designs and functionalities by understanding the unique specific requirements of mobile app customers and stakeholders of enterprises in various industries. Implemented across multiple platforms such as Apple IOS, Android, Windows Phone, HTML and made available for publishing on established mobile app markets.



Figure 17:Mobile App Development

- Web-Based Developments

Professional web development services (Web-Application Development & Website Development) to corporate, business and professional organizations in the form of Outsourced Web Developments, ranging from brochure sites to composite multi-functional portals.

- Consulting Services



Figure 18:Consulting Services

Offering a full range of consulting services with consultants that will guide the clients from every step of the way. Whether it be a new project or enhancements on an existing

application our services are certified to deliver on-time and within budget results. Certified in offering the following services: Java Technology Practice MYSQL Outsourcing Practice and Microsoft Technologies.

1.2 Co – operator plan

1.2.1 Vision

“Make life easy” by using ICT.



Figure 19:Co – operator plan

1.2.2 Mission

We strive to achieve our vision by being dynamic and benchmarking ourselves against the best in the world. Continuing improvement through creativity and innovation is a way of life at hSenid. Our approach is to nurture and guide our employees to embrace this culture and unleash their potential as we endeavor to be leaders in our field.

1.2.3 Milestones

- The following Milestones indicate the heights reached by hSenid Group of companies, in leading the constantly evolving Software Industry.
- 1997 - Beginning of hSenid
- 1998 - First client and signing contract
- 1999 - First international accolade
- 2000 - Launch very first payroll system “ePayRoll”
- 2001 - First foreign investor of hSenid
- 2003 - Establishment of hSenid mobile
- 2005 - hSenid reached to Singapore
- 2006 - hSenid opening in Africa
- 2008 - hSenid joins hands with IPM
- 2010 - Establishment in india
- 2011 - The beginning of API at hSenid mobile “AppZone”
- 2013 - hSenid launches “PeoplesHR” in Australia
- 2015 - Launch hSenid entrepreneur incubator “Loft1024”
- 2015 - hSenid launches LNBTI

1.2.4 Values

Being a part of a family is the environment that has been created in the workplace which is based on the Sri Lankan culture. On the other hand, one would see a mixed culture at hSenid Software Lanka that is nourished with the Japanese business ethics, culture and management style.

1.2.5 Number of employees (hSenid Software Lanka)

- Total employees - 32
 - Permanent - 26
 - Probation - 2
 - Trainees - 4

1.3 Management practices

1.3.1 Leave information

- Leaves at hSenid Software Lanka are according to the Shop and Office Employees Act No.19 of 1954. Working hours are 8.30 to 5.30 including lunch hours 9 hours a day. Working days are weekdays Monday to Friday.
- There are several types of leaves,
 - Casual leaves
 - Authorized No pay leave
 - Unauthorized no pay leave
 - Maternity leaves
 - Annual leave
 - Election leaves

Note: Interns have only ½ day casual leave for a month.

- 8 statutory holidays and Poya days are which are mentioned in the shop and office employee act are paid holiday. And also Saturdays and Sundays are offices holidays. If the employee had to work on that holiday can get a day off within 2 weeks.

1.3.2 EPF and ETF

- Permanent employees receive EPF according to the Employees' Provident Fund Act No.15 of 1958.
- ETF is received according to the Employees' Trust Fund Act No.46 of 1980.

1.3.3 Company offers

- hSenid Group guarantees the optimum processing power, highest quality standards, timely delivery and confidentiality for all development needs. State-of-the-art development facilities, working experience from around the globe and the flexibility to work in multinational & multicultural teams have given hSenid Group a unique advantage over other industry competitors.

1.3.4 Recruitment Procedure

There are several recruitment methods in hSenid Software International. Both internal and external.

- Promotions
- Transfer
- Internal advertising through notice board and Emails
- External advertising through the website, Facebook, LinkedIn and Instagram
- Internship programs - Summer Internship program

In the selection process job applicants has to face 3-4interviews.

- The first Interview is with team- members in the same department. It will be with technology lead or architect.
- The Second Interview is with the department manager.
- If an applicant passes these interviews, he/she has to face interview with HR
- Then finally the applicant will get a chance to have an interview with the CEO.

Then after reviewing the profile of the job applicant one more time will inform the applicant through call and email.

1.3.5 Security policy

- VPN

Maintain secure, encrypted tunnels with demanding customers of overseas and local.

1.3.6 Compensation

- The main compensation is a basic salary. And if work on night shift or special project overseas will add an allowance. All compensations will transfer to the employees' bank account on 28 of every month.

Chapter 2 - Training Experience

2.1 Introduction

As an industrial trainee, I joined hSenid Software International 20th of Jul 2020. I worked 6 months until 20 of January 2021. I nearly completed 27 working weeks at hSenid Software International. I was assigned to hSenid Software Lanka Japan Department as a Trainee Software Engineer.

Within these 6 months, the first 2 months were the training only period to get familiarized with the company environment and technologies. From 3rd month, when we planned the day, we planned 50% for the training plan and another 50%, assigning for the internal project. If there is an unplanned task with high priority, we were planning the day according to the priority based.

2.2 My Training Experience and Actions Performed

On the first day of my training period, Tech lead had a morning meeting with me and other trainees, and introduced our supervisor to us. After that he did a small brief introduction about what kind of tasks and training plan assigned to us within 6 months. Tech Lead recommended us to maintain our daily diary properly and get the remarks from the supervisor as recommended.

Until he shared our training plan with us, he advised us to self-study about Curl. He advised us to maintain documentation for each task as a blog.

So, I will share my Curl blog here as additional material done by me.



command line tool and library
for transferring data with URLs

Figure 20: CURL

Curl is a command line tool and library for transferring data with URL's. Curl supports for almost every protocol. Ex: HTTP, HTTPS, SMTP, IMAP etc. Curl can be used to send and receive data between client and server.

Install Curl

For Ubuntu- sudo apt install curl
 For Mac OS - brew install curl

For windows - Windows 10 or above version default installation or use binary file

Basic Usage
 Response
 curl <URL>

This basic syntax fetches the content from a given URL.

Ex: curl <http://www.hsenid.com/>

```
c:\Users\USER\Downloads>curl http://www.hsenid.com/
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en-gb" lang="en-gb" dir="ltr">
    <head>
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <!-- head -->
        <base href="http://www.hsenid.com/" />
        <meta http-equiv="content-type" content="text/html; charset=utf-8" />
        <meta name="keywords" content="HR Software, human resource software, hsenid software, HRIS, mobile software providers, HRIS, HRM software, HRO, HR outsourcing, mobile applications for the enterprise, HR capabilities" />
        <meta name="generator" content="Joomla! - Open Source Content Management" />
        <title>hSenid.com | HR Software, Mobile Software Provider|HRIS - HR Software Providers</title>
        <link href="/hsenidnew/images/logo-circle.png" rel="shortcut icon" type="image/vnd.microsoft.icon" />
        <link href="/hsenidnew/components/com_sppagebuilder/assets/css/font-awesome.min.css" rel="stylesheet" type="text/css" />
        <link href="/hsenidnew/components/com_sppagebuilder/assets/css/animate.min.css" rel="stylesheet" type="text/css" />
        <link href="/hsenidnew/components/com_sppagebuilder/assets/css/sppagebuilder.css" rel="stylesheet" type="text/css" />
        <link href="/hsenidnew/components/com_sppagebuilder/assets/css/sppagecontainer.css" rel="stylesheet" type="text/css" />
        <link href="http://www.hsenid.com/hsenidnew/components/com_layer_slider/base/static/layerslider/css/layerslider.css?ver=6.5.8" rel="stylesheet" type="text/css" />
        <link href="http://fonts.googleapis.com/css?family=Lato:100,300,regular,700,900&subset=Indie+Flower:regular%7COSwald:300,regular,700%7CRobot" rel="stylesheet" type="text/css" />
        <link href="/hsenidnew/components/com_sppagebuilder/assets/css/magnific-popup.css" rel="stylesheet" type="text/css" />
        <link href="/hsenidnew/templates/probe/css/bootstrap.min.css" rel="stylesheet" type="text/css" />
        <link href="/hsenidnew/templates/probe/css/font-awesome.min.css" rel="stylesheet" type="text/css" />
        <link href="/hsenidnew/templates/probe/css/legacy.css" rel="stylesheet" type="text/css" />
        <link href="/hsenidnew/templates/probe/css/template.css" rel="stylesheet" type="text/css" />
        <link href="/hsenidnew/templates/probe/css/presets/preset1.css" rel="stylesheet" type="text/css" class="preset" />
        <link href="/hsenidnew/templates/probe/css/custom.css" rel="stylesheet" type="text/css" />
        <link href="/hsenidnew/plugins/system/helix3/assets/css/pagebuilder.css" rel="stylesheet" type="text/css" />
        <link href="/hsenidnew/templates/probe/css/frontend-edit.css" rel="stylesheet" type="text/css" />
        <style type="text/css">
.scrollup, .scrollup:hover {
    background-color: #0099f0;
}
#spbp-addon-152961169280 .spbp-addon-title { margin:0px 0px 0px 0px; padding:0px 0px 0px; }
font-size:40px; line-height:40px;
#spbp-addon-1506296158383 .spbp-addon-title { (display:inline-block);
font-size:40px; line-height:40px;
#spbp-addon-152961169214 .spbp-addon-title { (display:inline-block);
font-size:40px; line-height:40px;
#spbp-addon-1506296158388 .spbp-addon-title { (display:inline-block);
```

Figure 21: Install Curl1

```
C:\Users\USER\Downloads>curl http://www.hsenid.com/
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en-gb" lang="en-gb" dir="ltr">
<head>
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
        <!-- head -->
        <base href="http://www.hsenid.com/" />
    <meta http-equiv="content-type" content="text/html; charset=utf-8" />
    <meta name="keywords" content="HR Software,Human resource software,hSenid software,HRIS,mobile software providers,HRIS,HRM software, HRO,HR outsourcing" />
    <meta name="description" content="hSenid is a software development company specialized in human resource software and mobile applications for the enter
table HR capabilities" />
    <meta name="generator" content="Joomla! - Open Source Content Management" />
    <title>hSenid.com | HR Software,Mobile Software Provider|HRM Software|HRIS - HR Software Provider</title>
    <link href="/hsenidnew/images/logo-circle.png" rel="shortcut icon" type="image/vnd.microsoft.icon" />
    <link href="/hsenidnew/components/com_sppagebuilder/assets/css/font-awesome.min.css" rel="stylesheet" type="text/css" />
    <link href="/hsenidnew/components/com_sppagebuilder/assets/css/animate.min.css" rel="stylesheet" type="text/css" />
    <link href="/hsenidnew/components/com_sppagebuilder/assets/css/sppagebuilder.css" rel="stylesheet" type="text/css" />
    <link href="/hsenidnew/components/com_sppagebuilder/assets/css/sppagecontainer.css" rel="stylesheet" type="text/css" />
    <link href="http://www.hsenid.com/hsenidnew/components/com_layer_slider/base/static/layerslider.css?ver=6.5.8" rel="stylesheet" type="t
    <link href="http://fonts.googleapis.com/css?family=latof:100,300,regular,700,900&subset=CIndie+Flower:regular%7Coswald:300,regular,700%CRobot
    <style type="text/css" />
    <link href="/hsenidnew/components/com_sppagebuilder/assets/css/magnific-popup.css" rel="stylesheet" type="text/css" />
    <link href="/hsenidnew/templates/probe/css/bootstrap.min.css" rel="stylesheet" type="text/css" />
    <link href="/hsenidnew/templates/probe/css/font-awesome.min.css" rel="stylesheet" type="text/css" />
    <link href="/hsenidnew/templates/probe/css/legacy.css" rel="stylesheet" type="text/css" />
    <link href="/hsenidnew/templates/probe/css/template.css" rel="stylesheet" type="text/css" />
    <link href="/hsenidnew/templates/probe/css/presets/preset1.css" rel="stylesheet" type="text/css" class="preset" />
    <link href="/hsenidnew/templates/probe/css/custom.css" rel="stylesheet" type="text/css" />
    <link href="/hsenidnew/plugins/system/helix3/assets/css/pagebuilder.css" rel="stylesheet" type="text/css" />
    <link href="/hsenidnew/templates/probe/css/frontend-edit.css" rel="stylesheet" type="text/css" />
    <style type="text/css">
.scrollup, .scrollup:hover {
    background-color: #009ff0;
#spbb-addon-1529661169280 .spbb-addon-title {margin:0px 0px 0px 0px;padding:0px 0px 0px 0px;}}
font-size:40px;line-height:40px;
#spbb-addon-1506296158383 .spbb-addon-title { display:inline-block;}
font-size:40px;line-height:40px;
#spbb-addon-1529661169214 .spbb-addon-title { display:inline-block;}
font-size:40px;line-height:40px;
#spbb-addon-1506296158382 .spbb-addon-title { display:inline-block;}
```

Figure 22:Install Curl2

Download file with Curl

`curl -o <filename. filetype > <URL>`

Curl directly downloads the URL content and without prints it to the terminal, save the output as a file with a given name.

Ex: curl -o abc.txt <https://www.facebook.com>

```
C:\Users\USER\Downloads>curl -o abc.txt https://www.facebook.com
 % Total      % Received % Xferd  Average Speed   Time     Time      Current
               Dload  Upload Total   Spent    Left Speed
100  201k     0  201k     0      0  201k      0  --:--:--  0:00:01  --:--:--  167k
```

C:\Users\USER\Downloads>

Figure 23:curl -o abc.txt https://www.facebook.com

```

<!DOCTYPE html>
html lang="en" id="facebook" class="no_js">
head>meta charset="utf-8" />meta name="referrer" content="default" id="meta_referrer" /><script nonce="bnQ18e0q">window._cstart=new Date();</script><script nonce="bnQ18e0q">function
envFlush(a){function b(b){for(var c in a)b[c]=a[c]}window.requireLazy?window.requireLazy(["Env"],b):(window.Env=window.Env||[],b(window.Env))}envFlush
({!ajaxpipe_token:"AXjfJirtfz_NKbgk","timeslice_heartbeat_config":{"pollIntervalMs":33,"idleGapThresholdMs":60,"ignoredTimesliceNames":{"requestAnimationFrame":true,"Event listenHandler mousemove":true,"Event listenHandler mouseover":true,"Event listenHandler mouseout":true,"Event listenHandler scroll":true,"isHeartbeatEnabled":true,"isArtilleryOn":false}},"shouldLogCounters":true,"timeslice_categories":
{"react_render":true,"reflow":true}, "sample_continuation_stacktraces":true, "do_mutation_flag":true, "stack_trace_limit":30, "timesliceBufferSize":5000, "show_invariant_decoder":false, "compat_iframe_token":"A05gc73QkY0ahl_K", "isQuick":false});</script><style nonce="bnQ18e0q">_DEV_=oCavalryLogger>window.CavalryLogger||function(a)
{this.lid=a,this.is_metric_collected=1>this.is_detailed_profiler=1>this.instrumentation_started=1>this.pagelet_metrics={},this.events={},this.ongoing_watch={}
},this.values={t_cstart:window._cstart},this.piggy_values={},this.bootloader_metrics={},this.resource_to_pagelet_mapping={},this.initializeInstrumentation&this.initializeInstrumentation()
,CavalryLogger.prototype.setIsDetailedProfiler=function(a){this.is_detailed_profiler=a};return this},CavalryLogger.prototype.setTTIEvent=function(a){this.tti_event=a;return
this},CavalryLogger.prototype.setValue=function(a,b,c,d){d=d?this.piggy_values[this.values[(typeof d)[a]]=="undefined"]||{}&(d[a]=b);return
this},CavalryLogger.prototype.getLastTtIValue=function(){return this.lastTtIValue},CavalryLogger.prototype.setTimeStamp=function(a,b,c,d){(this.mark
(a);var e=this.values.t_cstart||[this.values.t_start];e[d]=CavalryLogger.now();this.setTtIEvent&this.tti_event&(this.lastTtIValue=e),this.setTimeStamp
("t_tti",b));return this},CavalryLogger.prototype.mark=typeof console=="object"&&console.timestamp?function(a){console.timestamp(a)}:function()
{},CavalryLogger.prototype.addPiggyback=function(a,b){this.piggy_values[a]=b;return this},CavalryLogger.instances={},CavalryLogger.disableArtilleryOnUntilOffLogging=1
,CavalryLogger.getInstance=function(a){typeof a==="undefined"&&(a=CavalryLogger.id);CavalryLogger.instances[a]||(CavalryLogger.instances[a]=new CavalryLogger(a));return
CavalryLogger.instances[a]},CavalryLogger.setPageID=function(a){if(CavalryLogger.id==a){(var b=CavalryLogger.getInstance());CavalryLogger.instances[a]=b;CavalryLogger.instances
[a].lid=a;delete CavalryLogger.instances[0];CavalryLogger.id=a};CavalryLogger.now=function(){return

```

Figure 24:Figure 23:curl -o abc.txt https://www.facebook.com2

HTTP request/response

HTTP makes a HTTP request to the server, the server responds back with a HTTP response, which

contains the content of that page. HTTP requests contain the request method, URL, some headers, and

some data as part of the request body. Data is optional, depending on request method type.

Most common types

the request methods are “GET” and “POST”.

Typically, we use “GET” requests to retrieve information from the server, and “POST” to submit data to the server with given data.

I used sample rest APIs from <https://reqres.in/> site to check GET, POST, PUT AND DELETE command.

GET method

Get the response from GET request method

Curl <URL>

Ex: curl <https://reqres.in/api/users/2>

```

C:\Users\USER\Downloads>curl https://reqres.in/api/users/2
curl: (6) Could not resolve host: reqres.in

C:\Users\USER\Downloads>curl https://reqres.in/api/users/2
{"data":{"id":2,"email":"janet.weaver@reqres.in","first_name":"Janet","last_name":"Weaver","avatar":"https://s3.amazonaws.com/uifaces/faces/twitter/josephstein/128.jpg","ad":{"company
":"http://statuscode.org/","text":"A weekly newsletter focusing on software development, infrastructure, the server, performance, and the stack end of things."}}
C:\Users\USER\Downloads>

```

Figure 25:Get the response from GET request method

Get the response and headers using GET request method

If we use I flag, we can get headers also. Not only the response.

Curl -i <URL>

EX: curl -i <https://reqres.in/api/users/2>

```
C:\Users\USER\Downloads>curl -i https://reqres.in/api/users/2
HTTP/1.1 200 OK
Date: Thu, 23 Jul 2020 11:40:39 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 371
Connection: keep-alive
Set-Cookie: cfid=dc7ca4e4461ca4df144709d913d49809f1595504439; expires=Sat, 22-Aug-20 11:40:39 GMT; path=/; domain=.reqres.in; HttpOnly; SameSite=Lax; Secure
X-Powered-By: Express
Access-Control-Allow-Origin: *
Etag: W/"173-1Z8bn0V7Ujh6FK7NCaza1ccWXQk"
Via: 1.1 vegur
Cache-Control: max-age=14400
CF-Cache-Status: HIT
Age: 6670
Accept-Ranges: bytes
cf-request-id: 041d12ca6d000019e94c823200000001
Expect-CT: max-age=604800, report-uri="https://report-uri.cloudflare.com/cdn-cgi/expect-ct"
Server: cloudflare
CF-RAY: 5b7520bd7f5d19e9-SIN

{"data": {"id": 2, "email": "janet.weaver@reqres.in", "first_name": "Janet", "last_name": "Weaver", "avatar": "https://s3.amazonaws.com/uifaces/faces/twitter/josephstein/1
https://statuscode.org/", "text": "A weekly newsletter focusing on software development, infrastructure, the server, performance, and the stack end of things."}}
```

Figure 26: Get the response and headers using GET request method

POST method

Use POST method to send data. We can post data on the server if we use POST method. It contains header

with -H flag and data in the method body.

Curl -X POST -H <header> <URL> <data>

```
curl -X POST -H "Accept: application/json" https://reqres.in/api/users -d  
"name=morpheus&job= leader"
```

```
C:\Users\USER\Downloads>curl -X POST -H "Accept:application/json" https://reqres.in/api/users -d "name=morpheus&job= leader  
C:\Users\USER\Downloads>
```

Figure 27:POST method

PUT method

Use POST method to update data on the server.

```
curl -X POST -H "Accept: application/json" https://reqres.in/api/users -d  
        {"name=morpheus&job= zion  
             resident"}  
        
```

```
C:\Users\USER>curl -X POST -H "Accept: application/json" https://reqres.in/api/users -d {"name=morpheus&job= zion resident"} [{"name":"morpheus","job": "zion resident"}, {"id": "630", "createdAt": "2020-07-24T03:56:24.683Z"}]
C:\Users\USER>
```

Figure 28:PUT method

We can get headers also if we use the i flag.

```
curl -X POST -H "Accept: application/json" https://reqres.in/api/users -d
      {"name=morpheus&job= zion
                   resident"}  
{"name=morpheus&job= zion
                   resident"}
```

```
C:\Users\USER>curl -i -X POST -H "Accept: application/json" https://reqres.in/api/users -d {"name=morpheus&job= zion
ident"}
HTTP/1.1 201 Created
Date: Fri, 24 Jul 2020 03:59:38 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 94
Connection: keep-alive
Set-Cookie: __cfduid=d596784c246841027e4b800e80d6e5d4a1595563177; expires=Sun, 23-Aug-20 03:59:37 GMT; path=/; domain
eqres.in; HttpOnly; SameSite=Lax; Secure
X-Powered-By: Express
Access-Control-Allow-Origin: *
Etag: W/"5e-1I4nUrTnb0MJHanTxgdmA0FBW9I"
Via: 1.1 vegur
CF-Cache-Status: DYNAMIC
cf-request-id: 0420930fe7000001ee123b6200000001
Expect-CT: max-age=604800, report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"
Server: cloudflare
CF-RAY: 5b7abac63c4201ee-SIN

{"name":"morpheus", "job": " zion resident", "id": "178", "createdAt": "2020-07-24T03:59:38.241Z"}  

C:\Users\USER>
```

Figure 29:i flag.

Delete method

Using a given specific user, we can delete using the DELETE method.

Curl -X DELETE <https://reqres.in/api/users/2>

```
C:\Users\USER>
C:\Users\USER>
C:\Users\USER>Curl -X DELETE https://reqres.in/api/users/2  

C:\Users\USER>
```

Figure 30:Delete method

After the Curl studies, Tech Lead shared a google style sheet, which includes all the trainee tasks which we have to complete as a trainee. Our first task was, go through each task and assign estimated time for each task.

After that, the supervisor did an introduction which we are going to use within our training period and explained all internal projects and client ongoing projects, which currently exists inside the Japan unit of hSenid.

Because of the Corona pandemic situation, we couldn't go to the office and work there, we got instructions to work remotely from home and to use our personal computers. We used a Skype application for meeting calls to connect with office people.

Following is my training plan designed by our Tech Lead. It consists of every point, what he expects from us when following this plan.

Training plan link:

<https://docs.google.com/spreadsheets/d/1cNTKwGl9tknPkJby4iC43sY1NOeQVP5bAESsB0PUUi0/edit#gid=1132470953>

When we start to continue one of these tasks, we have to update the excel sheet shared by Tech Lead and mentioned as “On Progress”. If we completed each task, we have to change the status into “Done”.

Item	Training	Description	# days	progress	Status	Remarks
1	Introduction					
2	Set up Linux environment	Setup office accounts and required software (can use package manager software)	1	1	Done	
3	Linux basic training	Linux Tutorial: http://www.ee.surrey.ac.uk/Teaching/Unix/index.html Bash Tutorial: https://help.ubuntu.com/community/Beginners/BashScripting	2	2	Done	finished both linux and bash tutorial
4	Set up Java	multiple jdk's (8,11,14) and should be able to change effective jdk version via terminal command	1	1	Done	Installed Jdk 8 and Jdk 11 LTS versions using tar . now able to switch between many versions
5	Set up MySQL	Should be installed using tar file and should not start at system boot	1	1	Done	When install my sql using tar.. i got a temporary password. When i try to change that password,i got an error called : "error while loading shared libraries: libcurls.so.5." After installing libcurls5, i was able to change the password successfully.
6	Understanding Git	Learn Basic git commands and setup git for local pc. * create and save ssh key (optional) * maintain repository * learn pull,shelf,commit,push, tag, branch, merge, etc... and use them	2	3	Done	https://docs.google.com/document/d/1zRNCo9dXLDo29wxgXoV1bLwXSj6NNTrJ8_pV7Sp3Pl/edit?usp=sharing
7	Set up IntelliJ Idea and understand basic features	Setup intelliJ idea community edition using tar	1	1	Done	
		Following is required, * jdk versions knowledge (rough idea on features) * OOP concepts * Flow management (conditional and looping) * java stream api				

Figure 31:training plan.

This is our training plan.

Set up Linux environment
 Setup office accounts and required software (can use package manager software)

Linux basic training
 Linux Tutorial: <http://www.ee.surrey.ac.uk/Teaching/Unix/index.html> Bash Tutorial:
<https://help.ubuntu.com/community/Beginners/BashScripting> (Beginners BashScripting, n.d.)

Set up Java
 multiple JDKs (8,11,14) and should be able to change effective JDK version via terminal command

Set up MySQL

Should be installed using tar file and should not start at system boot

Understanding Git

Learn Basic git commands and setup git for local pc.
 create and save SSH key (optional)
 maintain repository
 learn pull, shelf, commit, push, tag, branch, merge, etc... and use them (Learning basic Git commands, n.d.).

Set up IntelliJ Idea and understand basic features

Setup IntelliJ idea community edition using tar

OOP concepts and Java core concepts
Following is required,
JSK versions knowledge (rough idea on features)
OOP concepts
Flow management (conditional and looping)
java stream API
lambda functions and default methods in interfaces
IO (NIO)
Basics of using threads and concurrent programming
>> create a system for a bank (should be tracked in git. no need of a DB connection)
>>> user idea to debug the sample application using debug points (debug inside of a lambda)

Tomcat training 1

setup and learn how to deploy a web app using tomcat (How to Install Tomcat 9 on Ubuntu 18.04, n.d.).
>>> how to enable debugging for a webapp using idea and tomcat

JSP understanding

Sample project using JSP (the UI for the bank app)
>>> use MVC structure

Maven

Setup and understand maven (convert the sample project (sample-bank) to use maven
should be able to build using maven
add 3rd party dependencies
run using maven

Software Unit Testing 1

write test cases to verify bank-app logic (not required to use Junit but better if possible)
JavaFX
Basic understanding of FX
>>> create calculator application

Software Testing 1

write manual test cases for the calculator/bank app and execute, record them with results

Spring training

Create a spring web application (develop bank application as a spring webapp. use xml configuration)
>>> use proper packaging
>>> extract API Layer for the UI to be connected

Software Unit Testing 2

Junit (4/5) understanding,
>>> write test cases in Junit and execute them to get a final report
>>> skip tests using maven when required
>>> get basic understanding of mocking (use Mockito)

Hibernate

basic understanding and knowledge on how to use in a spring app
>>> integrate MYSQL DB to the bank spring application

>>> add test cases for the repository layer (repository should be added as a separate maven module)

Spring Boot Training Basic 1

Understanding of spring boot and starters

>>> change the banking app to be a spring boot webapp (use annotation-based configuration)

>>> use separate rest API layer to connect UI to app (Springboot build a RESTful web API, n.d.)

>>> use spring transactions to control data persistence and rollback (Transactional annotation)

>>> understand how to deploy spring boot application without server container

>>> how to debug a spring boot application using idea

>>> starting spring boot app in terminal

Software Unit Testing 3

>>> writing test cases for the rest API layer Test Driven Development

>>> get an understanding of TDD concepts

>>> add hire-purchase module to bank app (use TDD approach)

Spring Boot Training Basic 2

Learn about microservices and monolithic. Extract the logic as a separate Rest API service for the bank app (there will be 2 services after this)

>>> change the UI deployed to call the Rest API using http

API Testing

Use Postman to test the rest API (record the results)

Frontend development: Angular

Study basics of angular framework (latest version is ok) and create the frontend for the bank app

>>> user the rest API to execute the logic

Web deployment – 1

Study about NGINX and deploy the angular frontend with it

Spring Security

enable spring security and secure the backend API.

>>> use JWT to handle authorization (backend should issue a token valid for 30 min and frontend should refresh the token once its expired)

>>> modify the rest layer test cases to accommodate these changes

Web deployment – 2

Study docker and containers.

>>> containerize both frontend and backend services and deploy them in testbed

As our very first task, the supervisor advised us to install the UBUNTU operating system for our personal computers in dual boot way. Ubuntu is the OS, which is used within hSenid company as their main working environment.

For OS installation, we assigned a one-day time period and we were able to complete within one day as planned. I went through YouTube tutorials as references. After the installation of Ubuntu, my next task was to learn basic Linux commands. Supervisor shared two materials with us to cover basic Linux and bash commands. We estimated and assigned 2 days for that task. After the completion of that task, the supervisor gave us a simple quiz to check our basic knowledge. With this task, I was able to Familiar with Linux environment and

commands, then learned to write shell scripts.

I will add my self-designed Linux and Bash tutorial here.

UNIX is an operating system which was first developed in the 1960s, and has been under constant development ever since.

It is a stable, multi-user, multi-tasking system for servers, desktops and laptops.

UNIX systems also have a graphical user interface (GUI) similar to Window. But knowledge of UNIX is required for operations which aren't covered by a graphical program

UNIX consists of 3 parts.

Kernel Shell Programmes

Kernel is the hub of UNIX OS. It allocates time and memory to programs and handles the file store and communications in response to system calls.

Shell acts as an interface between user and kernel.

Everything in UNIX is either a file or a process.

A process is an executing program identified by a unique PID (process identifier). A file is a collection of data. They are created by users using text editors, running compilers etc.

Directory Structure

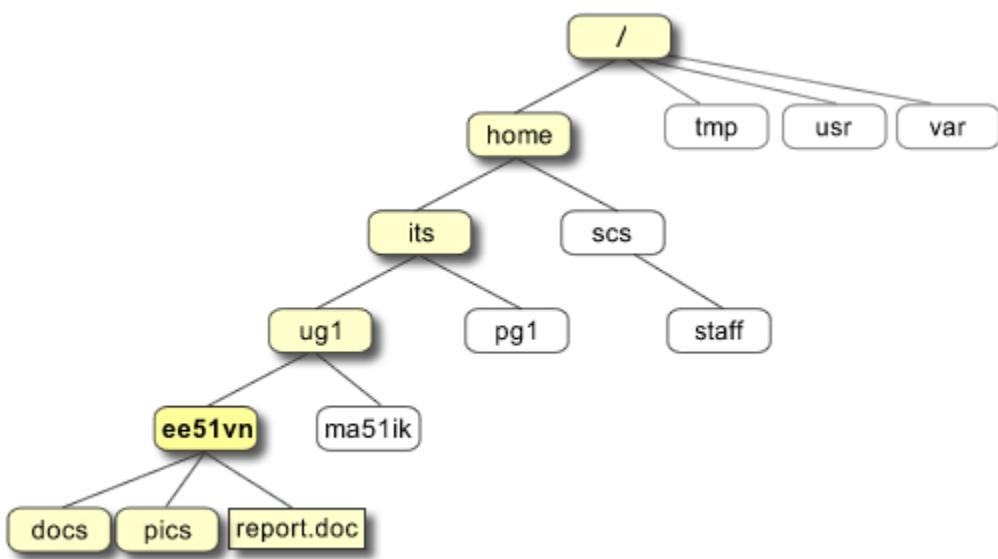


Figure 32: Directory Structure

```
asith@asith-X542UQR:~$ ls
Desktop  Downloads  Music      Public       snap       Videos
Documents  hsenid    Pictures   read_file.sh  Templates
asith@asith-X542UQR:~$
```

Figure 33:lowercase L and lowercase S

The **ls** command (lowercase L and lowercase S) lists the contents of your current working directory.

```
asith@asith-X542UQR:~$ ls -a
.           .config   .java     Public          .thunderbird
..          Desktop   .local    read_file.sh  Videos
.bash_history  Documents .mozilla  snap
.bash_logout   Downloads Music    .ssh
.bashrc        .gnupg   Pictures  .sudo_as_admin_successful
.cache        hsenid   .profile  Templates
asith@asith-X542UQR:~$
```

Figure 34:To list all files in your home directory

To list all files in your home directory including those whose names begin with a dot, type
Known as hidden files.

ls -a - list all files and directories
ls ~/directory_name
Mkdir directory_name - create directory name
Cd directory_name - go inside directory
 cd .
cd .. - change to parent directory
Pwd - path of current directory

```

asith@asith-X542UQR:~$ mkdir asithtesting
asith@asith-X542UQR:~$ cd asithtesting
asith@asith-X542UQR:~/asithtesting$ ls =a
ls: cannot access '=_a': No such file or directory
asith@asith-X542UQR:~/asithtesting$ ls -a
. ..
asith@asith-X542UQR:~/asithtesting$ cd .
asith@asith-X542UQR:~/asithtesting$ cd ..
asith@asith-X542UQR:~$ pwd
/home/asith
asith@asith-X542UQR:~$ cd asithtesting
asith@asith-X542UQR:~/asithtesting$ pwd
/home/asith/asithtesting
asith@asith-X542UQR:~/asithtesting$ ls asithtesting
ls: cannot access 'asithtesting': No such file or directory
asith@asith-X542UQR:~/asithtesting$ cd ..
asith@asith-X542UQR:~$ ls asithtesitng
ls: cannot access 'asithtesitng': No such file or directory
asith@asith-X542UQR:~$ ls asithtesting
asith@asith-X542UQR:~$ ls ~/asithtesting
asith@asith-X542UQR:~$ cd asith
bash: cd: asith: No such file or directory
asith@asith-X542UQR:~$ cd /asith
bash: cd: /asith: No such file or directory
asith@asith-X542UQR:~$ ls
asithtesting Documents hsenid Pictures read_file.sh Templates
Desktop Downloads Music Public snap Videos
asith@asith-X542UQR:~$ cd hsenid
asith@asith-X542UQR:~/hsenid$ ls ~/hsenid
shell 'ubuntu tasks'
asith@asith-X542UQR:~/hsenid$ █

```

Figure 35:Mkdir directory

```

asith@asith-X542UQR:~/Downloads$ cd ..
asith@asith-X542UQR:~$ ls
asithtesting Documents hsenid Pictures read_file.sh Templates
Desktop Downloads Music Public snap Videos
asith@asith-X542UQR:~$ cd hsenid
asith@asith-X542UQR:~/hsenid$ cp /home/asith/asithtesting/asith.txt .
asith@asith-X542UQR:~/hsenid$ ls
asith.txt shell 'ubuntu tasks'
asith@asith-X542UQR:~/hsenid$ █

```

Figure 36:copy files from one directory to another directory

Copy files from one directory to another directory

Move files from one directory to another

```
asith@asith-X542UQR:~/asithtesting$ ls
asith.txt
asith@asith-X542UQR:~/asithtesting$ cd ..
asith@asith-X542UQR:~$ ls
asithtesting Documents hsendid Pictures read_file.sh Templates
Desktop Downloads Music Public snap Videos
asith@asith-X542UQR:~$ cd Documents
asith@asith-X542UQR:~/Documents$ ls
asith@asith-X542UQR:~/Documents$ mv
mv: missing file operand
Try 'mv --help' for more information.
asith@asith-X542UQR:~/Documents$ mv /home/asith/asithtesting/asith.txt .
asith@asith-X542UQR:~/Documents$ ls
asith.txt
asith@asith-X542UQR:~/Documents$ cd /asithtesting
bash: cd: /asithtesting: No such file or directory
asith@asith-X542UQR:~/Documents$ cd /home/asith/asithtesting
asith@asith-X542UQR:~/asithtesting$ ls
asith@asith-X542UQR:~/asithtesting$ █
```

Figure 37:Move files from one directory to another

Cat command.

```
asith@asith-X542UQR:~/Documents$ ls
asith.txt
asith@asith-X542UQR:~/Documents$ cat asith.txt
```

asith ishantha madurawala. this is my first editor.

Figure 38:Cat command.

head command for printing the first 10 lines.
tail command for print last 10 lines

grep command
Find the specified word from the file.

```
asith@asith-X542UQR:~/Documents$ grep madurawala asith.txt
asith ishantha madurawala. this is my first editor.
asith@asith-X542UQR:~/Documents$
```

Figure 39:grep command

-i use to ignore case sensitive

```
asith@asith-X542UQR:~/Documents$ grep -i ASITH asith.txt
asith ishantha madurawala. this is my first editor.
asith@asith-X542UQR:~/Documents$
```

Figure 40:-i use to ignore case sensitive

cp - copy file
mv- move file
rm - remove file
rmdir- remove directory
cat- print content in terminal
clear- clear the terminal
head- print first 10 lines
tail- print last 10 lines
grep- find specific word

Cat redirection

Without giving the specific file name, we type cat command, it reads the standard keyboard.

```
asith@asith-X542UQR:~/Documents$ cat
this is unix standard input command , when press return key. it will print in terminal
this is unix standard input command , when press return key. it will print in terminal
asith@asith-X542UQR:~/Documents$
```

Figure 41:Cat redirection

Redirecting the output

We use > symbol to redirect output commands. After pressing enter, the user can give keyboard input and it will save it to the specific file.

Then cat <filename>. It will print all content inside the file.

```
asith@asith-X542UQR:~/Documents$ cat>list1
toyota
mazda
honda
suzuki
nissan
asith@asith-X542UQR:~/Documents$ cat list1
toyota
mazda
honda
suzuki
nissan
asith@asith-X542UQR:~/Documents$
```

Figure 42: Redirecting the output

Using >> symbol, we can do more changes to file.

```
asith@asith-X542UQR:~/Documents$ cat >> list1
we can do more changes to the file with >> symbol. it will append to the context which
already stored in list1 file.
asith@asith-X542UQR:~/Documents$ cat list1
toyota
mazda
honda
suzuki
nissan
we can do more changes to the file with >> symbol. it will append to the context which
already stored in list1 file.
asith@asith-X542UQR:~/Documents$
```

Figure 43: Using >> symbol, we can do more changes to file.

Users can join two files using cat command. First, I will create another file called list2.

We can use cat <file1> <file2> >new_list_name to combined many files.

```
asith@asith-X542UQR:~/Documents$ cat> list2
mango
orange
pineapple
apple
asith@asith-X542UQR:~/Documents$ cat list2
mango
orange
pineapple
apple
asith@asith-X542UQR:~/Documents$ cat list1 list2> combinedlist
asith@asith-X542UQR:~/Documents$ cat combinedlist
cat: combinedlist: No such file or directory
asith@asith-X542UQR:~/Documents$ cat combinedlist
toyota
mazda
honda
suzuki
nissan
we can do more changes to the file with >> symbol. it will append to the context which
already stored in list1 file.
mango
orange
pineapple
apple
asith@asith-X542UQR:~/Documents$
```

Figure 44: Users can join two files using cat command

Sort

Using sort, the user can sort file content into alphabetically or numerically order.

```
asith@asith-X542UQR:~/Documents$ sort
1
5
54
666
7980
132
0
0
1
132
5
54
666
7980
asith@asith-X542UQR:~/Documents$
```

Figure 45: Sort

Using < symbol, we can set to get input from the file.

```
asith@asith-X542UQR:~/Documents$ sort < list2
apple
mango
orange
pineapple
asith@asith-X542UQR:~/Documents$
```

Figure 46: Using < symbol, we can set to get input from the file.

When users do commands, they have to output into a temporary file and use it as a second command . it takes time. So using pipe “|” user can combine both commands

Example: command 1- who command2- sort

We can join both of them using pipe.

```
asith@asith-X542UQR:~/Documents$ who
asith    :0          2020-07-30 10:59 (:0)
asith@asith-X542UQR:~/Documents$ who > people
asith@asith-X542UQR:~/Documents$ sort < people
asith    :0          2020-07-30 10:59 (:0)
asith@asith-X542UQR:~/Documents$
```

Figure 47:who

```
asith@asith-X542UQR:~/Documents$ who | sort
asith    :0          2020-07-30 10:59 (:0)
asith@asith-X542UQR:~/Documents$
```

Figure 48:Who / sort

Who | sort

Gave same result quicker and cleaner

Find a letter from sorted list 1 list2

```
asith@asith-X542UQR:~/Documents$ cat list1 list2 | sort | grep a
apple
honda
mango
mazda
nissan
orange
pineapple
toyota
we can do more changes to the file with >> symbol. it will append to the context which
already stored in list1 file.
asith@asith-X542UQR:~/Documents$
```

Figure 49:Cat >> list

Cat > list

Cat >> list

Sort <list

Who | sort

Cat file1 file2 > big file

Wild cards

Ls list* - filter words starts with list word

Ls *list - filter words finish with list word

```

asith@asith-X542UQR:~/Documents$ ls
asith.txt combinedlist list1 list2 people
asith@asith-X542UQR:~/Documents$ ls list*
list1 list2
asith@asith-X542UQR:~/Documents$ ls *t
asith.txt combinedlist
asith@asith-X542UQR:~/Documents$ 

```

Figure 50:Ls list* - filter words starts with list word

Ls ?ist -? Represent exactly one character

```

asith@asith-X542UQR:~/Documents$ ls ?eople
people
asith@asith-X542UQR:~/Documents$ 

```

Figure 51:Ls ?ist -? Represent exactly one character

Man command - online manual command

If the user doesn't know about the command. He can simply use man <command> to get information about the command.

Whatis keyword also give the same output.

If the user isn't sure about the keyword, the user can use **apropos** keyword to guess the keyword.

```

asith@asith-X542UQR:~/Documents$ apropos copy
cp (1)           - copy files and directories
cpgr (8)         - copy with locking the given file to the password or group file
cpio (1)         - copy files to and from archives
cppw (8)         - copy with locking the given file to the password or group file
dd (1)           - convert and copy a file
debconf-copydb (1) - copy a debconf database
gvfs-copy (1)    - (unknown subject)
install (1)      - copy files and set attributes
mcopy (1)        - copy MSDOS files to/from Unix
ntfscp (8)       - copy file to an NTFS volume.
objcopy (1)      - copy and translate object files
rcp (1)          - OpenSSH secure file copy
rsync (1)        - a fast, versatile, remote (and local) file-copying tool
scp (1)          - OpenSSH secure file copy
ssh-copy-id (1)   - use locally available keys to authorise logins on a remote mac..
x86_64-linux-gnu-objcopy (1) - copy and translate object files

```

Figure 52:apropos keyword

* - *list(linkedlist)

? - ?sith (asith)

Man - online manual

Whatis - online manual

Apropos - guess not sure keywords

File access rights

```
asith@asith-X542UQR:~/Documents$ ls -l
total 20
-rw-rw-r-- 1 asith asith 114 30 12:36 asith.txt
-rw-rw-r-- 1 asith asith 180 30 15:45 combinedlist
-rwxrwxr-x 1 asith asith 151 30 15:24 list1
-rwxrwxr-x 1 asith asith 29 30 15:44 list2
-rw-rw-r-- 1 asith asith 44 30 16:03 people
asith@asith-X542UQR:~/Documents$
```

Figure 53:ls -l

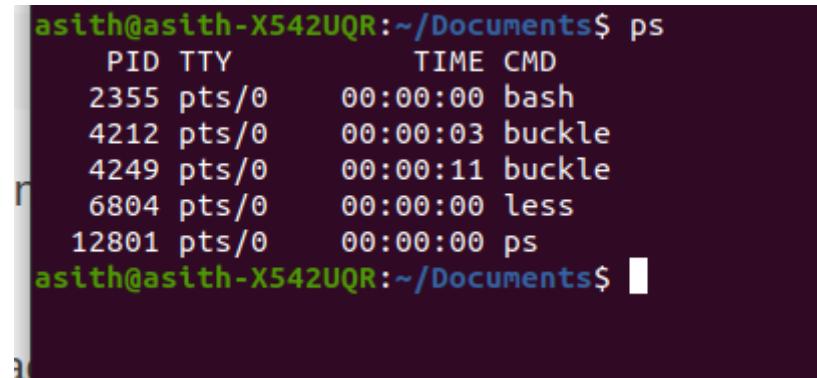
Symbol	Meaning
u	user
g	group
o	other
a	all
r	read
w	write (and delete)
x	execute (and access directory)
+	add permission
-	take away permission

give execute permission for people

```
asith@asith-X542UQR:~/Documents$ chmod oug+x people
asith@asith-X542UQR:~/Documents$ ls -l
total 20
-rw-rw-r-- 1 asith asith 114 30 12:36 asith.txt
-rw-rw-r-- 1 asith asith 180 30 15:45 combinedlist
-rwxrwxr-x 1 asith asith 151 30 15:24 list1
-rwxrwxr-x 1 asith asith 29 30 15:44 list2
-rwxrwxr-x 1 asith asith 44 30 16:03 people
asith@asith-X542UQR:~/Documents$
```

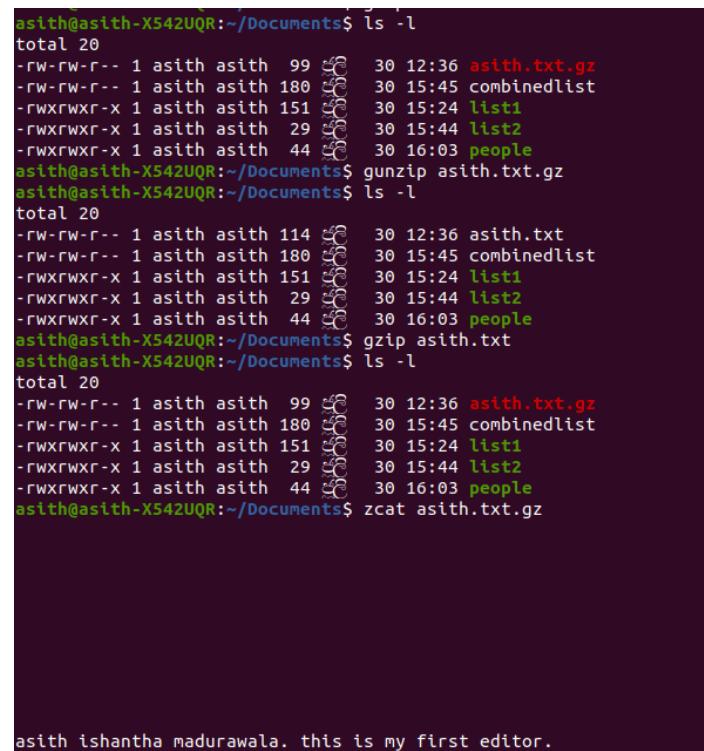
Figure 54:chmod oug+x people

To see information about your process, users can use PS command.



```
asith@asith-X542UQR:~/Documents$ ps
  PID TTY      TIME CMD
 2355 pts/0    00:00:00 bash
 4212 pts/0    00:00:03 buckle
 4249 pts/0    00:00:11 buckle
 6804 pts/0    00:00:00 less
12801 pts/0    00:00:00 ps
asith@asith-X542UQR:~/Documents$
```

Figure 55:ps



```
asith@asith-X542UQR:~/Documents$ ls -l
total 20
-rw-rw-r-- 1 asith asith  99 30 12:36 asith.txt.gz
-rw-rw-r-- 1 asith asith 180 30 15:45 combinedlist
-rwxrwxr-x 1 asith asith 151 30 15:24 list1
-rwxrwxr-x 1 asith asith  29 30 15:44 list2
-rwxrwxr-x 1 asith asith  44 30 16:03 people
asith@asith-X542UQR:~/Documents$ gunzip asith.txt.gz
asith@asith-X542UQR:~/Documents$ ls -l
total 20
-rw-rw-r-- 1 asith asith 114 30 12:36 asith.txt
-rw-rw-r-- 1 asith asith 180 30 15:45 combinedlist
-rwxrwxr-x 1 asith asith 151 30 15:24 list1
-rwxrwxr-x 1 asith asith  29 30 15:44 list2
-rwxrwxr-x 1 asith asith  44 30 16:03 people
asith@asith-X542UQR:~/Documents$ gzip asith.txt
asith@asith-X542UQR:~/Documents$ ls -l
total 20
-rw-rw-r-- 1 asith asith  99 30 12:36 asith.txt.gz
-rw-rw-r-- 1 asith asith 180 30 15:45 combinedlist
-rwxrwxr-x 1 asith asith 151 30 15:24 list1
-rwxrwxr-x 1 asith asith  29 30 15:44 list2
-rwxrwxr-x 1 asith asith  44 30 16:03 people
asith@asith-X542UQR:~/Documents$ zcat asith.txt.gz

asith ishantha madurawala. this is my first editor.
```

Figure 56:ls

History

Users can use the history command to see the history of commands the user typed.
Can use! <command> to see the old typed commands

```

asith@asith-X542UQR:~/Documents$ !sudo
sudo apt install quota
Reading package lists... Done
Building dependency tree
Reading state information... Done
quota is already the newest version (4.05-1).
The following packages were automatically installed and are no longer required:
  fonts-dejavu-extra java-common libatk-wrapper-java libatk-wrapper-java-jni
  libice-dev libpthread-stubs0-dev libsm-dev libxi-dev libxau-dev libxcb1-dev
  libxdmcp-dev libxt-dev x11proto-core-dev x11proto-dev xorg-sgml-doctools xtrans-dev
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 216 not upgraded.
asith@asith-X542UQR:~/Documents$ !grep
grep -i ASITH asith.txt
grep: asith.txt: No such file or directory
asith@asith-X542UQR:~/Documents$ 

```

Figure 57:!sudo

I prepared mind maps also for remind myself of the theories.

I will add a bash scripting self-designed report here.

Bash scripting is best compared to batch scripting.

Bash is very flexible and have many advanced features

Using bash language, we will learn to do lot of things in both GUI and CLI(command line interface)

Something is more achieved than others (Beginners BashScripting, n.d.). Ex: setting permission for directoreis are easy in CLI than GUI

Bash used not only to run programs and applications, it also used to write programs and scripts.

Creating folders

Deleting files

Deleting folders and their sub-folders

Opening applications as root

Backing up your files

Backing up your folders

Checking system performance

Check Devices

Checking wireless connection

Creating folders and creating files inside folder, then delete file

```
asith@asith-X542UQR:/$ cd /home/asith/asithtesting/bashtesing  
asith@asith-X542UQR:~/asithtesting/bashtesing$ cat> asith.txt  
i am creating a file insdie in bashtesing directory  
asith@asith-X542UQR:~/asithtesting/bashtesing$ cd /  
asith@asith-X542UQR:/$ rm /home/asith/asithtesting/bashtesing/asith.txt  
asith@asith-X542UQR:/$
```

Figure 58: `cat>asith.txt`

rm -rf <folder path>

This command will delete folders and subfolders in this folder.

This command will be deleting everything in your computer root directory

```
rm -rf /*  
rm -rf /
```

Only system administrators have permission to access system directories. Like
usr/local/bin.

Because it will affect all the systems. Trying to copy something into this folder will be
forbidden by default.

To run this, we have to use **sudo** command, to run command as root.

gksudo is the same as *sudo* but should be used to open any graphical applications as root
while *sudo* is intended for executing single commands.

Backing up your files.

cp -r (recursive) option

```
cp -r /directory/to/be/copied/ /where/to/copy/to/
```

Checking system performance

If your computer starts to lag, you can see which applications are using the most CPU
power with this command:

```
top
```

check Devices

Lsusb - to check whether, USB is connected or not.

Lspci - to check whether RAM is properly connected or not.

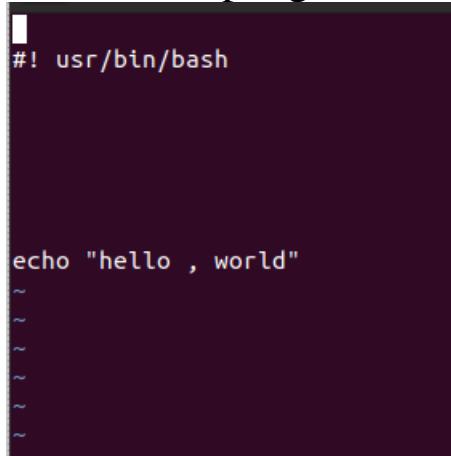
Show network Information

Ipaddr - to check whether, ipaddr

Show wireless information

Iwconfig - to check about wireless connection

Scripting



```
#!/usr/bin/bash

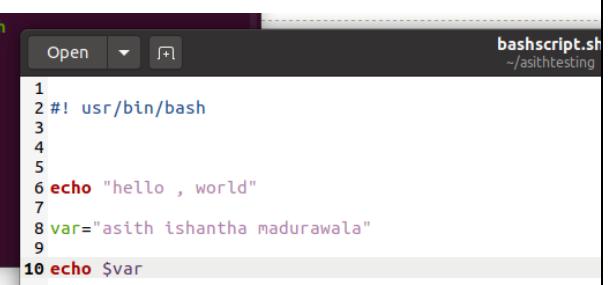
echo "hello , world"
~
```

Figure 59:`#!/usr/bin/bash`

```
asith@asith-X542UQR:~$ ls
asithtesting  Documents  hsenid  Pictures  read_file.sh  Templates
Desktop       Downloads  Music   Public    snap          Videos
asith@asith-X542UQR:~$ cd asithtesting
asith@asith-X542UQR:~/asithtesting$ vi bashscript.sh
asith@asith-X542UQR:~/asithtesting$ chmod a+x chmod.sh
chmod: cannot access 'chmod.sh': No such file or directory
asith@asith-X542UQR:~/asithtesting$ chmod a+x bashscript.sh
asith@asith-X542UQR:~/asithtesting$ ls -a
.  ..  asith.txt  bashscript.sh  bashtesing
asith@asith-X542UQR:~/asithtesting$ ls -l
total 8
-rw-rw-r-- 1 asith asith    0 31 10:08 asith.txt
-rwxrwxr-x 1 asith asith  44 31 10:49 bashscript.sh
drwxrwxr-x 2 asith asith 4096 31 10:15 bashtesing
asith@asith-X542UQR:~/asithtesting$ ./bashscript.sh
hello , world
asith@asith-X542UQR:~/asithtesting$
```

Figure 60: `./bashscript.sh`

```
-rwxrwxr-x 1 asith asith 44 31 10:49 bashscript.sh
drwxrwxr-x 2 asith asith 4096 31 10:15 bashtesing
asith@asith-X542UQR:~/asithtesting$ ./bashscript.sh
hello , world
asith@asith-X542UQR:~/asithtesting$ vi bashscript.sh
asith@asith-X542UQR:~/asithtesting$ ./bashscript.sh
hello , world
asith@asith-X542UQR:~/asithtesting$ vi bashscript.sh
asith@asith-X542UQR:~/asithtesting$ ./bashscript.sh
hello , world
asith ishantha madurawala
asith@asith-X542UQR:~/asithtesting$
```



The screenshot shows the contents of the `bashscript.sh` file in a vi editor. The file contains the following code:

```
1
2 #! /usr/bin/bash
3
4
5
6 echo "hello , world"
7
8 var="asith ishantha madurawala"
9
10 echo $var
```

Figure 61: `vi bashscript.sh`

```

asith@asith-X542UQR:~/asithtesting$ ls -l
total 8
-rw-r--r-- 1 asith asith 0 31 10:08 asith.txt
-rwxrwxr-x 1 asith asith 44 31 10:49 bashscript.sh
drwxrwxr-x 2 asith asith 4096 31 10:15 bashtesting
asith@asith-X542UQR:~/asithtesting$ ./bashscript.sh
hello , world
asith@asith-X542UQR:~/asithtesting$ vi bashscript.sh
asith@asith-X542UQR:~/asithtesting$ ./bashscript.sh
hello , world
asith@asith-X542UQR:~/asithtesting$ vi bashscript.sh
asith@asith-X542UQR:~/asithtesting$ ./bashscript.sh
hello , world
asith ishantha madurawala
asith@asith-X542UQR:~/asithtesting$ ./bashscript.sh
hello , world
asith ishantha madurawala
what is your name:
asith ishantha madurawala
what is your company
hsenid
what is country
sri lanka
my name is asith ishantha madurawala and my company is hsenid and my country is
sri lanka
asith@asith-X542UQR:~/asithtesting$ 

```

Figure 62:./bashscript.sh

```

Please enter your name : asith
Please enter your age : 25
Please enter your sex. Male/Female : male
So you're a 25 year old male called asith
asith@asith-X542UQR:~/asithtesting$ 

```

Figure 63: bashscript program

If Statements

```

asith@asith-X542UQR:~/asithtesting$ ./bashscript.sh
Please enter a fruit name : mango
i don't like mango
asith@asith-X542UQR:~/asithtesting$ ./bashscript.sh
Please enter a fruit name : apple
i want mango
asith@asith-X542UQR:~/asithtesting$ 

```

Figure 64:If Statements bashscript

```

h@asith-X542UQR:~/asithtesting$ ./bashscript.sh
use enter a fruit name : pineapple
like pineapple
h@asith-X542UQR:~/asithtesting$ 

1 #! /usr/bin/bash
2
3
4
5 read -p "Please enter a fruit name : " fruit
6
7
8 if [ $fruit == mango ]
9 then echo "i don't like mango"
10 elif [ $fruit == apple ]
11 then echo "i want mango"
12 elif [ $fruit == pineapple ]
13 then echo "i like pineapple"
14
15
16
17
18
19
20
21 fi
22

```

Figure 65: If Statements bashscript

```

1 #! /usr/bin/bash
2
3
4
5 foovar= who
6 echo $foovar
7

```

Figure 66: #! /usr/bin/bash

```

#!/usr/bin/bash
}
}

; foovar= who | grep a
; echo $foovar
'

```

Figure 67: who | grep a

FUNctions

Users can write functions also.

```

2 #! /usr/bin/bash
3
4
5
6 echofunction (){
7
8 echo "this is my first function in bash"
9 echo
10 }
11
12 asithfunction (){
13
14 echo "i will write variable and print user output in this function"
15 echo
16
17 read -p "please enter your degree name: " degree_name
18
19 echo $degree_name
20 echo
21
22
23 }
24
25
26 echofunction;
27 echo
28 asithfunction;
29 echo
30 echo "i printed both functions here"
31 asithfunction;
32 echo

```

Figure 68:bashscript function

```

2 #! /usr/bin/bash
3
4
5 echo "please enter something in lower case. we will convert into UPPER CASE!!!!"
6
7 read abcd
8
9 var=$( echo $abcd | tr "[a-z]" "[A-Z]" )
10
11 echo $var
12
13

```

Figure 69:bashscript function function2

```

1 #! /usr/bin/bash
2
3
4 echo "type: i love bapan"
5
6 read country
7
8 var=$( echo $country | tr "[b]" "[j]" )
9
10 echo $var
11

```

Figure 70:SED

SED

sed is one of the most complicated scripting languages on the GNU / Linux system. I am only going to cover the *s/* command here.

```

1 #! /usr/bin/bash
2
3
4
5 read abc
6 echo $abc | sed "s/change/not_change/"
7

```

Figure 71:SEDI

Next task was to install Java JDK version 8,11 and 14.

Before the installation, Supervisor advised us to get an idea about JDK version 8 to 14 with their differences and features.

When search information regarding 7-14, previous versions (7,8,9) of JDK resources were available lot in internet. But JDK 10,11,12,13,14, information resources were little. Because updates come every 6 months and few changes only added.

When study about JDK, I got an idea about LTS version and minor releases. And learn how to jump between many java versions in computer according to use.

I created a self-study report also. I will submit it here.

Java JDK Java 7 Java7 called Dolphin It supports dynamic languages. We can use Strings in switch statements. Automatic resource management for try- catch statement. No need close manually. Simplified varargs method of declaration Single catch statements for throwing multiple exceptions. We can use underscores in numeric literals. Ex: 11_1111_234 Use diamond for generic instance creation. Ex: List<String>
--

Java 8

Java 1.8 is introduced in 2014. Java JDK 8 is a major LTR (Long Term Release). In previous we couldn't define method in Interfaces. We only able to do the declare it. We only can have abstract method in interfaces. There is a new feature in Java 8, we can define methods in our interface.

In previous version, we couldn't have internal iteration of collection API. But in Java 8, We can do it.

Main feature we got from 1.8 is, lambda expression. If we have a big statement, using lambda expression, you can convert into one line statement.

Ex:

```
Thread t= new Thread ()-> System.out.println("hello hsenid")
```

Summary:

We can define method in interface

We can use lambda expression

We can use stream API

Java 9

Java 9 is a Long-Term Support We got Jigsaw project in Java 9. To understand Jigsaw, we have to understand modularity, which is a new feature in Java 9. In java 9, introduced modules. Every project in java needs rt.jar file.

String class, System class... stored in this runtime.jar file. Depend on project, requirements are different. So, all of classes in .jar will not be used for each project. In Java 9, approximately 94 modules were created with classes. According to needs, we can have specific modules without having all of modules. There can be have module dependency also.

Second one is REPL (Read Eval Print Loop), It can have with java now same as other languages like python. No need to write many codes just print a line. We now can directly use print line without main and class.

Additionally, we can have Interfaces, which private methods defined in Java 9.

Java 10

Java 10 is a minor release. Only have few features.

We can use VAR key word to declare variable.

There are few changes to collector API.

Java 11

Lazy allocation of compiler threads.
Updated locale data to Unicode CLDR v33
HTTP client
Brainpool EC support

A No-Op garbage collector
Nest based access control
Transport Layer security
Collection.toArray (IntFunction) default method

Java 12

Syntactical changes-Switch Expression and String literal
Support for Unicode 11
Support for Compact Number Formatting
Square Character Support for Japanese New Era

Java 13

Added FileSystems.newFileSystem(Path, Map<String,>) Method
New java.nio.ByteBuffer Bulk get/put Methods Transfer Bytes Without Regard to Buffer Position
New Japanese Era Name Reiwa

Java 14

Helpful NullPointerExceptions
Switch Expressions

Preview features- pattern matching for instances of, Text blocks, Packaging tool

He advised us to use tar files when installing applications. It was hard for me in the beginning and I had so many troubles while installing software using binary files. I had to download those files, extract them, add a bin path to the bashrc file and execute them through commands, it was a new experience for me. Once I accidentally changed the bashrc file so my command line crashed.

My next task was to install MySQL using a tar file (How To Install MySQL on Ubuntu 18.04, n.d.). I will add all screenshots, which are related to installing MYSQL.

```
asith@asith-X542UQR:~/Downloads/test2/mysql$ sudo groupadd mysql
asith@asith-X542UQR:~/Downloads/test2/mysql$ sudo useradd -r -g mysql -s /bin/false mysql
asith@asith-X542UQR:~/Downloads/test2/mysql$ ls
mysql-5.7.31-linux-glibc2.12-x86_64.tar.gz
asith@asith-X542UQR:~/Downloads/test2/mysql$ sudo tar zxvf mysql-5.7.31-linux-glibc2.12-x86_64.tar.gz
mysql-5.7.31-linux-glibc2.12-x86_64/
mysql-5.7.31-linux-glibc2.12-x86_64/man/
mysql-5.7.31-linux-glibc2.12-x86_64/man/man1/
mysql-5.7.31-linux-glibc2.12-x86_64/man/man1/mysql_multi.i
mysql-5.7.31-linux-glibc2.12-x86_64/man/man1/comp_err.i
mysql-5.7.31-linux-glibc2.12-x86_64/man/man1/mysql.server.i
```

Figure 72:mysql new group

```
asith@asith-X542UQR:~/Downloads/test2/mysql$ cd /usr/local
asith@asith-X542UQR:/usr/local$ sudo tar zxvf /home/asith/Downloads/test2/mysql/mysql-5.7.31-linux-glibc2.12-x86_64.tar.gz
mysql-5.7.31-linux-glibc2.12-x86_64/
mysql-5.7.31-linux-glibc2.12-x86_64/man/
mysql-5.7.31-linux-glibc2.12-x86_64/man/man1/
mysql-5.7.31-linux-glibc2.12-x86_64/man/man1/mysqld_multi.1
mysql-5.7.31-linux-glibc2.12-x86_64/man/man1/comp_err.1
mysql-5.7.31-linux-glibc2.12-x86_64/man/man1/mysql.server.1
asith@asith-X542UQR:/usr/local/mysql$ sudo bin/mysqld --initialize --user=mysql
2020-07-31T08:30:48.848297Z 0 [Warning] TIMESTAMP with implicit DEFAULT value is deprecated. Please use --explicit_defaults_for_timestamp server option (see documentation for more details).
2020-07-31T08:30:51.350951Z 0 [Warning] InnoDB: New log files created, LSN=45790
2020-07-31T08:30:51.881476Z 0 [Warning] InnoDB: Creating foreign key constraint system tables.
2020-07-31T08:30:52.069605Z 0 [Warning] No existing UUID has been found, so we assume that this is the first time that this server has been started. Generating a new UUID: 24e963dd-d308-11ea-ac8f-9822ef00ce3d.
2020-07-31T08:30:52.127334Z 0 [Warning] Gtid table is used but not supported. Table 'mysql.gtid_executed' cannot be opened.
2020-07-31T08:30:52.980461Z 0 [Warning] CA certificate ca.pem is self signed.
2020-07-31T08:30:53.298356Z 1 [Note] A temporary password is generated for root@localhost: #NXpLV+uo8EX
asith@asith-X542UQR:/usr/local/mysql$ mysql -u root -p#NXpLV+uo8EX
asith@asith-X542UQR:/usr/local/mysql$ sudo /etc/init.d/mysql status
● mysql.service - LSB: start and stop MySQL
   Loaded: loaded (/etc/init.d/mysql; generated)
   Active: active (running) since Fri 2020-07-31 14:14:37 +0530; 32min ago
     Docs: man:systemd-sysv-generator(8)
 Process: 11176 ExecStart=/etc/init.d/mysql start (code=exited, status=0/SUCCESS)
   Tasks: 28 (limit: 9351)
  Memory: 173.6M
    CGroup: /system.slice/mysql.service
            └─11200 /bin/sh /usr/local/mysql/bin/mysqld_safe --datadir=/usr/local/mysql/data --pid-file=/usr/local/...
               ├─11286 /usr/local/mysql/bin/mysqld --basedir=/usr/local/mysql --datadir=/usr/local/mysql/data --plugin...
...
31 14:14:36 asith-X542UQR systemd[1]: Starting LSB: start and stop MySQL...
31 14:14:36 asith-X542UQR mysql[11176]: Starting MySQL
31 14:14:37 asith-X542UQR mysql[11176]: . *
31 14:14:37 asith-X542UQR systemd[1]: Started LSB: start and stop MySQL.
asith@asith-X542UQR:/usr/local/mysql$
```

Figure 73:check mysql connection

```
asith@asith-X542UQR:/usr/local/mysql$ mysql -u root -p#NXpLV+uo8EX
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 3
Server version: 5.7.31

copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Figure 74:mysql create new user

After the successful installation of MYSQL, I moved to the next task called Git understanding and setup.

In Hsenid Software International PVT ltd, **GIT** is the most used open-source VCS ([version control system](#)) that allows us to track changes made to files.

Following points learnt through Gitlab understanding.

Through Gitlab understanding training tasks, I was able to learn basic commands like pull, shelf, commit, push, tag, branch and merge etc.

Learn how to use git commands to create a repository and maintain it with add, commit, push etc.

Learn how to use Gitlab remote repository. Learn how to create branches, checkout between many branches, learn how to access other members, programs files and do the changes without checkout their branch.

Learn the difference between local and remote.

Difference between download and cloan.

I will add my GitLab self-study report here.



Figure 75:gitlab github

GIT is the most widely used open-source VCS (**version control system**) that allows you to track changes made to files.

Companies and programmers usually use GIT to collaborate on developing software and applications.

A GIT project consists of three major sections: **the working directory**, **the staging area**, and **the git directory**.

The working directory is where you add, delete, and edit the files. Then, the changes are staged (indexed)

in the staging area. After you commit your changes, the snapshot of the changes will be saved into the git directory.

Basic GIT Commands

git init will create a new local GIT repository. The following Git command will create a repository in the current directory:

you can create a repository within a new directory by specifying the project name:

git init [project name]

git clone is used to copy a repository. If the repository lies on a remote server, use:
git clone username@host:/path/to/repository

run the following basic command to copy a local repository:
git clone /path/to/repository

git add is used to add files to the staging area. For example, the basic Git following command will index the temp.txt file:
git add <temp.txt>

git commit will create a snapshot of the changes and save it to the git directory.
git commit -m "Message to go with the commit here"

git config can be used to set user-specific configuration values like email, username, file format, and so on. To illustrate, the command for setting up an email will look like this:

git config --global user.email youremail@example.com

–global flag tells GIT that you're going to use that email for all local repositories. If you want to use different emails for different repositories, use the command below:

git config --local user.email youremail@example.com

git status displays the list of changed files together with the files that are yet to be staged or committed.

git status

git push is used to send local commits to the master branch of the remote repository.

Here's the basic code structure:

git push origin <master>

Replace <master> with the branch where you want to push your changes when you're not intending to push to the master branch.

git checkout creates branches and helps you to navigate between them. For example, the following basic command creates a new branch and automatically switches you to it:

command git checkout -b <branch-name>

To switch from one branch to another, simply use:

git checkout <branch-name>

git remote lets you view all remote repositories. The following command will list all connections along with their URLs:
git remote -v

To connect the local repository to a remote server, use the command below:

git remote add origin <host-or-remoteURL>

the following command will delete a connection to a specified remote repository:
git remote rm <name-of-the-repository>

git branch will list, create, or delete branches. For instance, if you want to list all the branches present in the repository, the command should look like this:
git branch

If you want to delete a branch, use:
git branch -d <branch-name>

git pull merges all the changes present in the remote repository to the local working directory.
git pull

git merge is used to merge a branch into the active one.
git merge <branch-name>

git diff lists down conflicts. In order to view conflicts against the base file, use
git diff --base <file-name>

The following basic command is used to view the conflicts between branches before merging them:

git diff <source-branch> <target-branch>

To list down all the present conflicts, use:
git diff

git log is used to see the repository's history by listing certain commit details. Running the command will get you an output that looks like this:

commit 15f4b6c44b3c8344caasdac9e4be13246e21sadw
Author: Alex Hunter <aalexh@gmail.com>
Date: Mon Oct 1 12:56:29 2016 -0600

git reset command will reset the index and the working directory to the last git commit state.

git reset --hard HEAD

git rm can be used to remove files from the index and the working directory.
git rm filename.txt

git stash command will temporarily save the changes that are not ready to be committed.

git stash

git show is a command used to view information about any git object.
git show

git fetch allows users to fetch all objects from the remote repository that don't currently reside in the local working directory.

git fetch origin

git ls-tree allows you to view a tree object along with the name, the mode of each item, and the blob's SHA-1 value. Let's say you want to see the HEAD, use:

git ls-tree HEAD

GitHub:Git repository practice

Configuring GitHub (Learning basic Git commands, n.d.)

After the successful installation, the user has to set up his/her configuration details. We can do that using the following two commands by changing username and user email that are used for creating a git hub account.

```
asith@asith-X542UQR:~$ git config --global user.name "asithishantha"  
asith@asith-X542UQR:~$ git config --global user.email "asithish075@gmail.com"
```

Figure 76:creating a git hub

After that, we can create a folder and it will work as a local repository, which can be pushed to a remote repository.

```
asith@asith-X542UQR:~$ git init lnbtihsenidasithtest  
Initialized empty Git repository in /home/asith/lnbtihsenidasithtest/.git/
```

Figure 77:pushed to a remote

After that, we can create a README file inside the folder. It will describe the project content.

```
asith@asith-X542UQR:~/lnbtihsenidasithtest$ gedit README  
asith@asith-X542UQR:~/lnbtihsenidasithtest$ ls  
README  
asith@asith-X542UQR:~/lnbtihsenidasithtest$ cat README  
this is my first repository. i am writing README file, whic explain the content  
of file.  
  
asith@asith-X542UQR:~/lnbtihsenidasithtest$ █
```

Figure 78:create a README file

After that we can create repository files inside the file.

"git add" command can be used to add any number of files and folders to the index. Here, when I say index, what I am referring to is a buffer like space that stores the files/folders that have to be added into the Git repository.

```
asith@asith-X542UQR:~/lnbtihsenidasithtest$ git add README
asith@asith-X542UQR:~/lnbtihsenidasithtest$ git add test.c
asith@asith-X542UQR:~/lnbtihsenidasithtest$
```

Figure 79:git add" command

all the files are added, we can commit it. This means that we can upload to the git repository.

Create a repository on GitHub. Notice that the name of the repository should be the same as the repositories on the local system. In this case, it will be "lnbtihsenidasithtest".

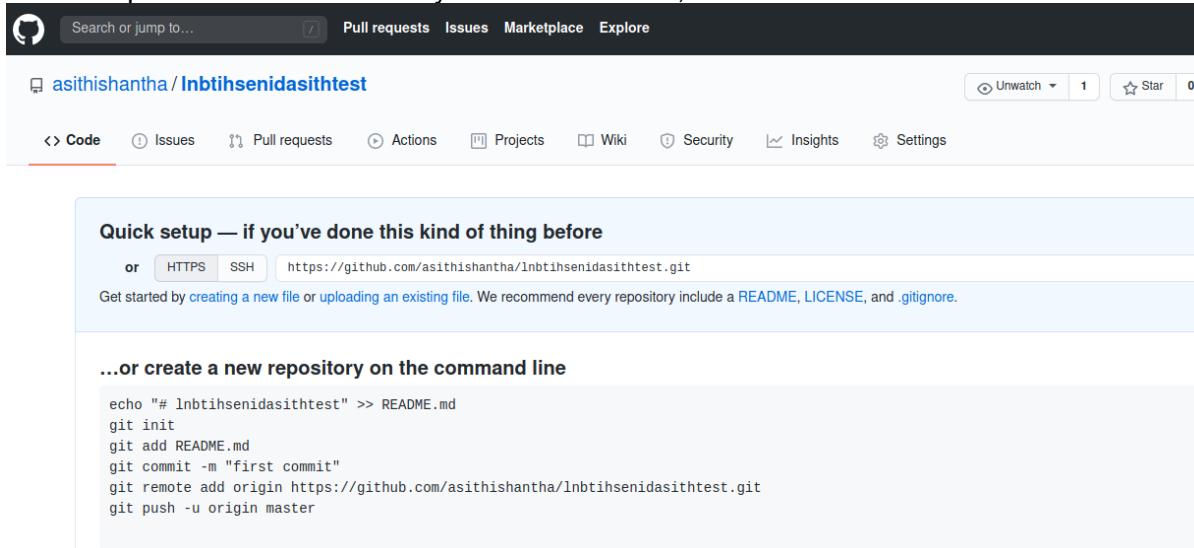


Figure 80:Create a repository on GitHub.

After creating a repository, we can push content from the local repository into the git repository.

final step is to push the local repository contents into the remote host repository (GitHub)

```
asith@asith-X542UQR:~/lnbtihsenidasithtest$ git remote add origin https://github.com/asithishantha/lnbtihsenidasithtest.git
asith@asith-X542UQR:~/lnbtihsenidasithtest$ git push origin master
Username for 'https://github.com':
```

Figure 81:to push the local repository contents into the remote host repository

```
asith@asith-X542UQR:~/lnbtihsenidasithtest$ git push origin master
Username for 'https://github.com': asithishantha
Password for 'https://asithishantha@github.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 458 bytes | 458.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0)
To https://github.com/asithishantha/lnbtihsenidasithtest.git
 * [new branch]      master -> master
asith@asith-X542UQR:~/lnbtihsenidasithtest$
```

Figure 82:git push origin master

After these commands, we can see the updated git repository.

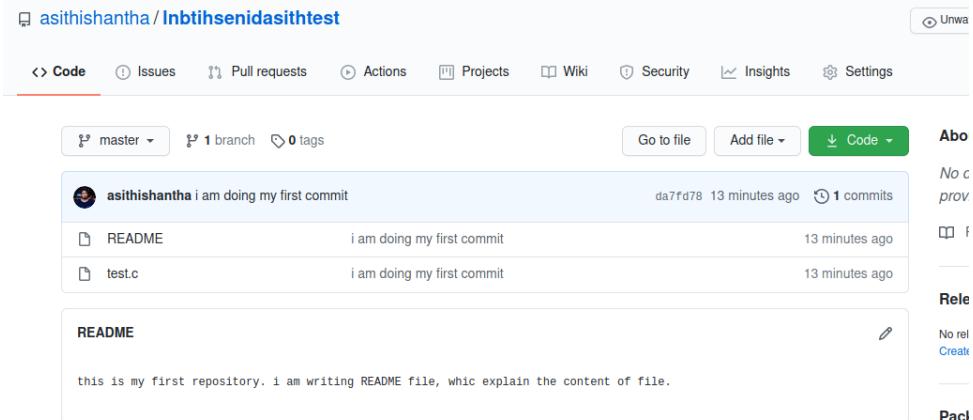


Figure 83:updated git repository.

Git-Lab(Industry use): Using Command line to PUSH to Git Lab

Use git add. in your bash to add all the files to the given folder.
If you want push only specific file, you can use git add <file_name>

you can create a commit message by git commit -m 'your message', which adds the change to the local repository.

```
asith@asith-X542UQR:~/gitlab-training-lnbti$ git add linkedlist.c
asith@asith-X542UQR:~/gitlab-training-lnbti$ git commit -m "linkedlist function: insert before node for linesr linked list"
[Asith_Ishantha ea873f3] linkedlist function: insert before node for linesr linked list
 1 file changed, 33 insertions(+)
 create mode 100644 linkedlist.c
```

Figure 84:by git commit -m 'your message',

Copy your remote repository's URL from git lab.

The HTTPS or URL is copied from the given git lab account, which is the place of the remote repository.

git remote add origin 'your_URL_name'

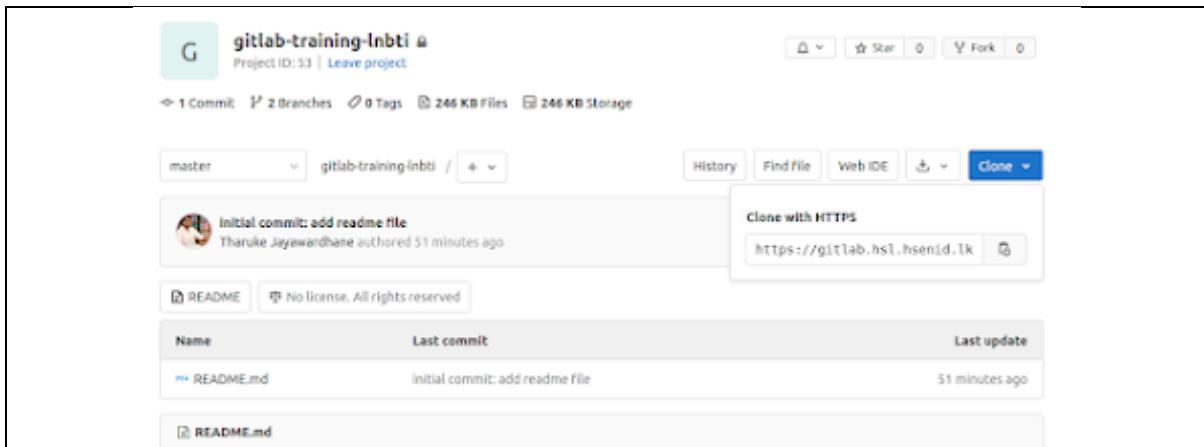


Figure 85:git remote add origin 'your_URL_name'

Push the code in your local repository to git lab

git push -u origin <master or branch name>
is used for pushing local content to git lab.

```
asith@asith-X542UQR:~/gitlab-training-lnbt$ git push
Username for 'https://gitlab.hsl.hsenid.lk': AsithIshantha
Password for 'https://AsithIshantha@gitlab.hsl.hsenid.lk':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 606 bytes | 606.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote:
remote: To create a merge request for Asith_Ishantha, visit:
remote:   https://gitlab.hsl.hsenid.lk/hsl-training/gitlab-training-lnbt/-/merge_requests?source_branch=Asith_Ishantha
remote:
To https://gitlab.hsl.hsenid.lk/hsl-training/gitlab-training-lnbt.git
 b7b18a4..ea873f3  Asith_Ishantha -> Asith_Ishantha
```

Figure 86:git push -u origin <master or branch name>

View your files in your repository hosted on git lab.

You can finally see the file hosted on git lab .

linkedlist function: insert before node for linesr linked list		
Asith Ishantha authored 18 minutes ago		ea873f3a
Name	Last commit	Last update
README	first commit	3 hours ago
README.md	first commit	1 hour ago
linkedist.c	linkedlist function: insert before node for linesr linked list	18 minutes ago
sampleprogram.c	i added a new graph structure program in c	2 hours ago

Figure 87:View your files in repository hosted on git lab.

Using Command line to PULL to Git lab

```

1 #include <stdio.h>
2 #include <stdlib.h>
3
4 // Define maximum number of vertices in the graph
5 #define N 6
6
7 // Data structure to store graph
8 struct Graph {
9     // An array of pointers to Node to represent adjacency list
10    struct Node* head[N];
11 };
12
13 // A data structure to store adjacency list nodes of the graph
14 struct Node {
15     int dest;
16     struct Node* next;
17 };
18
19 // data structure to store graph edges

```

```

^Z
[7]+ Stopped                  gedit sampleprogram.c
asith@asith-X542UQR:~/gitlab-training-lnbt$ 
asith@asith-X542UQR:~/gitlab-training-lnbt$ 
asith@asith-X542UQR:~/gitlab-training-lnbt$ 
asith@asith-X542UQR:~/gitlab-training-lnbt$ 
asith@asith-X542UQR:~/gitlab-training-lnbt$ gedit sampleprogram.c
Failed to register: GDBus.Error:org.freedesktop.DBus.Error.NoReply:
message bus without replying
[7]+ Killed                  gedit sampleprogram.c
asith@asith-X542UQR:~/gitlab-training-lnbt$ 
asith@asith-X542UQR:~/gitlab-training-lnbt$ gedit sampleprogram.c
asith@asith-X542UQR:~/gitlab-training-lnbt$ 
asith@asith-X542UQR:~/gitlab-training-lnbt$ 
asith@asith-X542UQR:~/gitlab-training-lnbt$ gedit sampleprogram.c
[7]+ 

```

Figure 88:Using Command line to PULL to Git lab

Even another developer committed my program. I didn't get the updated code. Because i still didn't pull the updated command from git remote repository. So, I will check out my branch and pull updated code.

```

~$ git pull
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 // Define maximum number of vertices in the graph
5 #define N 6
6
7 // Data structure to store graph
8 struct Graph {
9     // An array of pointers to Node to represent adjacency list
10    struct Node* head[N];
11    printf('Hello');
12 };
13
14 // A data structure to store adjacency list nodes of the graph
15 struct Node {
16     int dest;
17     struct Node* next;
18 };
19
20 // data structure to store graph edges

```

```

asith@asith-X542UQR:~/gitlab-training-lnbt$ git pull
Username for 'https://gitlab.hsl.hsenid.lk': AsithIshantha
Password for 'https://AsithIshantha@gitlab.hsl.hsenid.lk':
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://gitlab.hsl.hsenid.lk/hsl-training/gitlab-training-lnbt
   ea873f3..5798c72 Asith.Ishantha -> origin/Asith.Ishantha
 * [new branch]      master        -> origin/master
Updating ea873f3..5798c72
Fast-forward
 sampleprogram.c | 1 +
 1 file changed, 1 insertion(+)
asith@asith-X542UQR:~/gitlab-training-lnbt$ ls
linkedlist.c README README.md sampleprogram.c
[7]+ Killed                  gedit pull
asith@asith-X542UQR:~/gitlab-training-lnbt$ gedit sampleprogram.c
[7]+ 

```

Figure 89:I will check out my branch and pull updated code.

Git tag

```

asith@asith-X542UQR:~/gitlab-training-lnbt$ git tag
tagname
v1.0
asith@asith-X542UQR:~/gitlab-training-lnbt$ git tag --delete v1.0
Deleted tag 'v1.0' (was a7d30af)
asith@asith-X542UQR:~/gitlab-training-lnbt$ 
asith@asith-X542UQR:~/gitlab-training-lnbt$ 
asith@asith-X542UQR:~/gitlab-training-lnbt$ 
asith@asith-X542UQR:~/gitlab-training-lnbt$ 
asith@asith-X542UQR:~/gitlab-training-lnbt$ 
asith@asith-X542UQR:~/gitlab-training-lnbt$ 
asith@asith-X542UQR:~/gitlab-training-lnbt$ git status
git: 'statua' is not a git command. See 'git --help'.

The most similar command is
    status
asith@asith-X542UQR:~/gitlab-training-lnbt$ git status
HEAD detached at 1c6ef60
nothing to commit, working tree clean
asith@asith-X542UQR:~/gitlab-training-lnbt$ 

```

Figure 90:Git tag

After the Gitlab understanding training task, next task was OOP concepts and Java core concepts learning.

Following topics were the main topics to be covered through this task.

OOP concepts

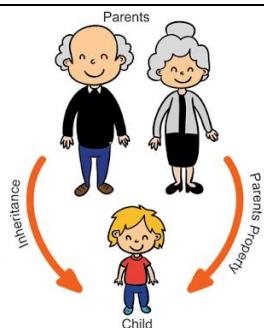


Figure 91: Inheritance is one of the key features of OOP

Inheritance is one of the **key features of OOP** (Object-oriented Programming) that allows us to define a new class from an existing class. For example,

```

class cal{
    public int Addval(int a, int b) {
        return a+b;
    }
}

class calVersion2 extends cal
{
    public int substra(int a, int b) {
        return a-b;
    }
}

class calLatestVersion extends calVersion2
{
    public int mul(int a, int b) {
        return a*b;
    }
}

```

Figure 92:Inheritance java program

In Java, we use the **extends** keyword to inherit from a class.
Here, we have inherited the **calLatestVersion** class from the **calVersion2** class.

The **calVersion2** is the super class (parent class or base class), and the **calLatestVersion** is a **subclass** (child class or derived class). The subclass inherits the fields and methods of the **super class**.

Example: Java Inheritance

```

public class Main {

    public static void main(String[] args) {
        System.out.println("this is a simple calculator");
        callLatestVersion obj= new callLatestVersion();
        int x= obj.mul( a: 2, b: 3);
        int y= obj.Addval( a: 2, b: 3);
        int z= obj.substraction( a: 2, b: 3);
        System.out.println("mul value is :" + x);
        System.out.println("add value is :" + y);
        System.out.println("substraction value is :" + z);

    }
}

```

Figure 93:Java Inheritance main program

Output

```

this is a simple calculator
mul value is :6
add value is :5
substraction value is :-1

```

Figure 94:Java inheritance output

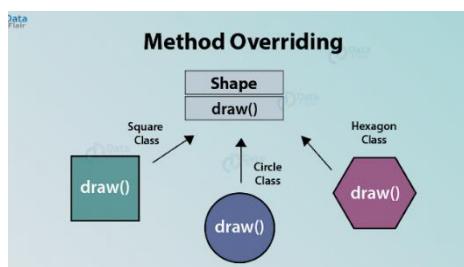


Figure 95:Java method overriding

In this post, we will learn about method overriding in Java with the help of examples

In the last [blog](#), we learned about inheritance. Inheritance is an OOP property that allows us to derive a new class (child class) from an existing class (parent class). The child class inherits the attributes and

methods of the parent class.

Now, if the same method is defined in both the parent class and the child class, then the method of the child class overrides the method of the parent class . This is known as method overriding.

Example 1: Method Overriding

```
class vehicle{  
  
    public void display_features(){  
        System.out.println("i have two seats and 4 wheels");  
    }  
  
}  
  
class car extends vehicle{  
    public void display_features(){  
  
        System.out.println("i have 5 seats and 4 wheels");  
    }  
  
}  
  
class bus extends vehicle{  
    public void display_features(){  
  
        System.out.println("i have 42 seats and 6 wheels");  
    }  
}
```

Figure 96:Java overriding program

main :

```

public class overriding {

    public static void main(String[] args){

        bus b1=new bus();
        car c1=new car();

        b1.display_features();

        c1.display_features();

    }

}

```

Figure 97:Overriding main class

Output:

```

i have 42 seats and 6 wheels
i have 5 seats and 4 wheels

Process finished with exit code 0

```

Figure 98:Output

Figure 99:overriding main class

In the above program, the **display_features()** method is present in both vehicle super class and the car, bus sub classes. When we call **display_features()** using the b1 or c1 objects (objects of the sub classes), the method inside the sub classes are called. The **display_features()** method of the sub classes override the same method of the super class.

Java Overriding Rules

Both the parent class and the child class must have the same method name, the same return type and the same parameter lists.

We cannot override the method declared as final and static.

We should always override abstract methods of the super class (will be discussed in later blogs).

super Keyword in Java Overriding

Can we access the method of the super class after overriding?

Well, the answer is **Yes**. To access the method of the super class from the subclass, we use the **super** keyword.

```
class vehicle{  
    public void display_features(){  
        System.out.println("i have two seats and 4 wheels");  
    }  
}  
  
class car extends vehicle{  
    public void display_features(){  
        super.display_features();  
        System.out.println("i have 5 seats and 4 wheels");  
    }  
}  
  
class bus extends vehicle{  
    public void display_features(){  
        super.display_features();  
        System.out.println("i have 42 seats and 6 wheels");  
    }  
}
```

Figure 100: Inheriting superclass subclass

Output:

```
i have two seats and 4 wheels  
i have 42 seats and 6 wheels  
i have two seats and 4 wheels  
i have 5 seats and 4 wheels
```

```
Process finished with exit code 0
```

Figure 101: Superclass subclass output

Inside **display_features()** of the car and bus sub classes, we have used **super.display_features()** to call **display_features() of the super class**.

Overriding Abstract Methods

In Java, abstract classes are created to be the super class of other classes. And, if a class contains an

abstract method, it is mandatory to override it.

We will learn more about abstract classes and overriding of abstract methods in later blogs.

I will share my self-designed note here.

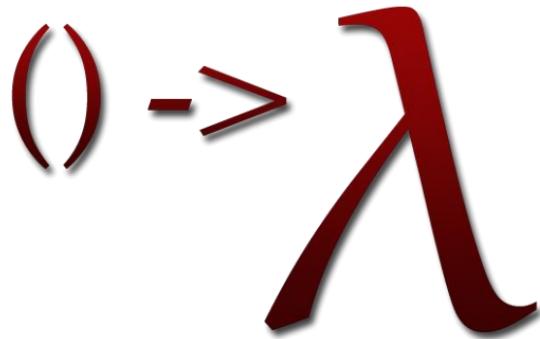


Figure 102:lambda expression

In this post, we will learn about Java lambda expression and the use of lambda expression with functional interfaces with the help of examples. The lambda expression was introduced for the first time in Java 8. Its main objective is to increase the expressive power of the language.

But, before getting into lambdas, we first need to understand functional interfaces.

What is Functional Interface?

If a Java interface contains one and only one abstract method (non implemented method) then it is termed as a functional interface. This only one method specifies the intended purpose of the interface.

```
@FunctionalInterface  
public interface myinterface {  
  
    // the single abstract method  
    int addValue();  
}
```

Figure 103:lambda functional interface

In the above example, the interface myInterface has only one abstract method addValue(). Hence, it is a functional interface. Here, we have used the annotation **@FunctionalInterface**. The annotation forces the Java compiler to indicate that the interface is a functional interface.

Hence, does not allow to have more than one abstract method. However, it is not compulsory.

Introduction to lambda expressions

- Lambda expression is, essentially, an anonymous or unnamed method. A function that doesn't have a name and doesn't belong to any class. The lambda expression does not execute on its own. Instead,
 - it is used to implement a method defined by a functional interface.

How to define lambda expressions in Java?

Here is how we can define lambda expressions in Java.

(parameter list) -> lambda body

The new operator (`->`) used is known as an arrow operator or a lambda operator.

As an example we have an abstract method like this which override inside abcd class.

```

interface myinterface
{
    // the single abstract method
    int addValue();
}

class abcd implements myinterface{

    @Override
    public int addValue() {
        return 150;
    }
}

public class abc {

    public static void main(String[] args){

        abcd obj = new abcd();
        int x=obj.addValue();
        System.out.println(x);

    }
}

```

Figure 104:define lambda expressions in Java

We can write this method using lambda expression as:

```

myinterface obj;
obj= () -> 150;

```

Figure 105:writing lambda expression in another way

Here, the method does not have any parameters. Hence, the left side of the operator includes an empty parameter. The right side is the lambda body that specifies the action of the lambda expression.

In this case, it returns the value 100.

Types of Lambda Body

In Java, the lambda body is of two types.

A body with a single expression

```
myinterface obj;  
obj= () -> System.out.println("lambda is very easy");
```

Figure 106:A body with a single expression

A body that consists of a block of code.

```
obj= () ->  
{  
    System.out.println("lambda is very easy");  
    System.out.println("A body that consists of a block of code.");  
  
};
```

Figure 107:a block body

This type of the lambda body is known as a block body. The block body allows the lambda body to include multiple statements. These statements are enclosed inside the braces and you have to add a semicolon after the braces.

Example: Lambda Expression

Let's write a Java program that returns the value 150 using the lambda expression. As mentioned earlier, a lambda expression is not executed on its own. Rather, it forms the implementation of the abstract method defined by the functional interface. So, we need to define a functional interface first

Note: this is the traditional implementation for abstract method

```

interface myinterface
{
    // the single abstract method
    int addValue();
}

class bcd implements myinterface{

    @Override
    public int addValue()
    {
        return 150;
    }
}

public class abc {

    public static void main(String[]args){

        myinterface obj;
        obj=new bcd();
        int x= obj.addValue();
        System.out.println(x);
    }
}

```

Figure 108:Single abstract method

Now i will use lambda expression to get the same output.

```
interface myinterface
{
    // the single abstract method
    int addValue();
}

public class abc {
    public static void main(String[] args){
        myinterface obj;
        obj= () -> 150;

        int x= obj.addValue();
        System.out.println(x);
    }
}
```

Figure 109:Lambda expression abstract method

Output:

```
150
```

```
Process finished with exit code 0
```

Figure 110:lambda expression output

Example 2:

```

interface myinterface
{
    // the single abstract method
    void addValue();
}

public class abc {

    public static void main(String[]args){
        myinterface obj;
        obj= () -> System.out.println("lambda is great");
        obj.addValue();
    }
}

```

Figure 111:lambda expression abstract program

Lambda Expressions with parameters

Till now we have created lambda expressions without any parameters. However, similar to methods,
lambda expressions can also have parameters. For example,

```

interface myinterface
{
    // the single abstract method
    int addValue(int b);
}

```

Figure 112:Lambda Expressions with parameters

This is the lambda implementation

```

myinterface obj;
obj= (a) -> {
    int total;
    int num=100;
    total= num+ a;
    return total;
};

int z=obj.addValue( b: 5);
System.out.println(z);

```

Figure 113:lambda implementation

Output:

105

Process finished with exit code 0

Figure 114: Lambda expression output



Figure 115:Java streams

The new [java.util.stream](#) package has been added to JDK8 which allows java developers to perform operations like search, filter, map, reduce, or manipulate collections like **Lists**. For example, we have a stream of data (in our case a **List** of **String**) where each string is a combination of country name and place of the country. Now, we can process this stream of data and retrieve only the places from Japan. For this, we can perform bulk operations in the stream by the combination of Stream API and Lambda expression.

Example: Demonstration of using lambdas with the Stream API

```

package com.company;
import java.util.ArrayList;
import java.util.List;

public class stream {

    // create an object of list using ArrayList
    static List<String> places = new ArrayList<>();

    // preparing our data
    public static List getPlaces() {

        // add places and country to the list
        places.add("sri lanka, colombo");
        places.add("japan, tokyo");
        places.add("India, Delhi");
        places.add("USA, New York");
        places.add("england, london");
        places.add("japan, osaka");

        return places;
    }

    public static void main( String[] args ) {

        List<String> myPlaces = getPlaces();
        System.out.println("Places from japan:");

        // Filter places from japan
        myPlaces.stream()
            .filter((p) -> p.startsWith("japan"))
            .map((p) -> p.toUpperCase())
            .sorted()
            .forEach((p) -> System.out.println(p));

    }
}

```

Figure 116:Array List Lambda expression

Output:

```

Places from japan:
JAPAN, OSAKA
JAPAN, TOKYO

```

Figure 117:Output lambda expression

In the above example, notice the statement,

```
// Filter places from Nepal
myPlaces.stream()
    .filter((p) -> p.startsWith("japan"))
    .map((p) -> p.toUpperCase())
    .sorted()
    .forEach((p) -> System.out.println(p));
```

Figure 118:lambda expression filter

Here, we are using the methods like **filter()**, **map()** and **forEach()** of the Stream API.

These methods can take a lambda expression as input. We can also define our own expressions based on the syntax we learned above. This allows us to reduce the lines of code drastically as we saw in the above example.

IO VS NIO

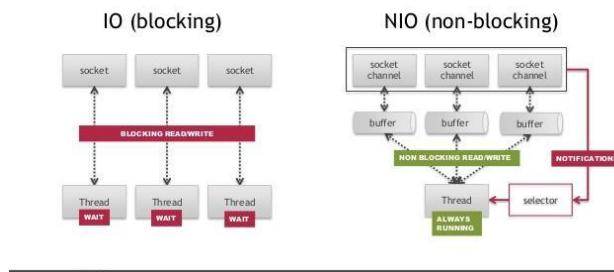


Figure 119:IS vs NIO

IO based on the Blocking I/O operation. NIO based on the Non-blocking I/O operation.

IO based on Stream-oriented and NIO based on buffer-oriented.

Channels are not available for IO operation and available for NIO operation.

Blocking I/O

Blocking I/O wait for the data to be write or read before returning. It means when the thread invoke a write() or read(), then the thread is blocked until there is some data available for read, or the data is fully written.

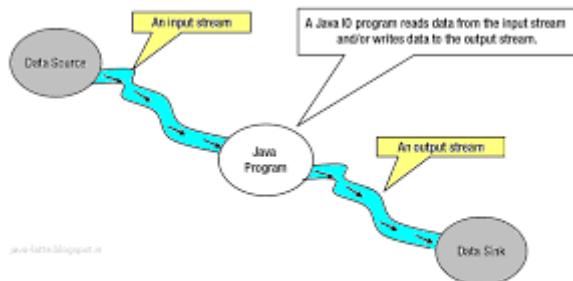
Non-blocking, I/O

non-Blocking, I/O no need to wait for the data to be written or read before returning. Java NIO non-blocking mode allows the thread to request writing data to a channel, but not wait for it to be fully written.

Stream Oriented vs. Buffer Oriented

Stream Oriented

Java I/O uses streams for transferring the data between a data source/sink and a java program. The I/O operation using this approach is slow.

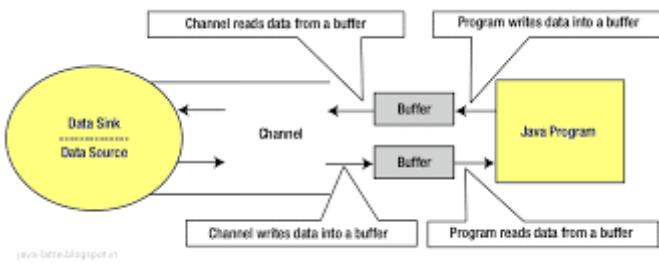


Flow of data using an input/output stream in a Java program

Figure 120: Stream Oriented

Buffer Oriented

Java NIO is buffer oriented I/O approach. Data is read into a buffer from which it is further processed using a channel. In NIO we deal with the channel and buffer for I/O operation.



Interaction between a channel, buffers, a Java program, a data source, and a data sink

Figure 121: Buffer Oriented

Main difference between stream and channel is,

Stream provides **one-way** and channel provides **two-way** transfer facility.

Channels

In Java NIO, the channel acts as a medium that transports the data efficiently between the entity and byte buffers. It reads the data from an entity and places it inside buffer blocks.

Channels act as a gateway provided by java NIO to access the I/O mechanism.

Selectors

Selector is a multiplexor of selectable channels, which is used as a special type of channel that can be put into non-blocking mode

It can examine one or more NIO Channels and determines which channel is ready for communication i.e. reading or writing.

What is the use of Selector?

Selector is used for handling many channels using one thread. So, we need less threads to handle the channel. Using threads is expensive for the operating system and increase of use selectors, makes efficiency of the system.

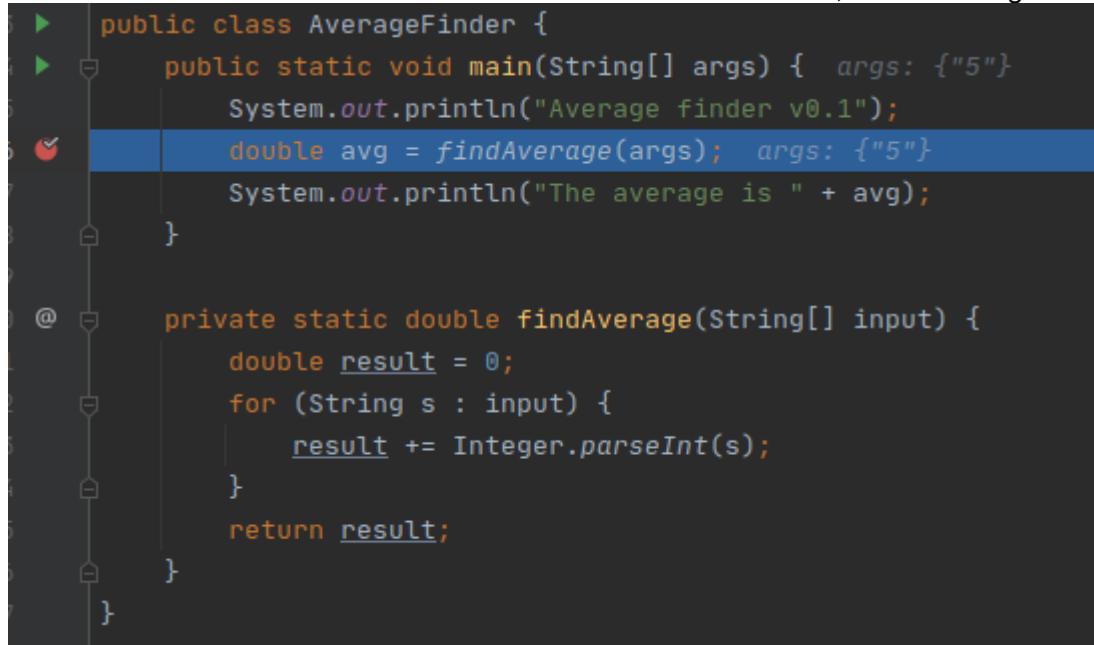
- Flow management (conditional and looping)
- Java stream API
- lambda functions and default methods in interfaces
- IO (NIO)

- Basics of using threads and concurrent programming
- >> create a system for a bank (should be tracked in git. no need of a DB connection)
- >>> user idea to debug the sample application using debug points (debug inside of a lambda)

Set breakpoints

Breakpoints indicate the lines of code where the program will be suspended for you to examine its state.

Click the Run button near the main method. From the menu, select Debug.



```
▶ public class AverageFinder {  
▶     public static void main(String[] args) { args: {"5"}  
▶         System.out.println("Average finder v0.1");  
▶         double avg = findAverage(args); args: {"5"}  
▶         System.out.println("The average is " + avg);  
▶     }  
▶  
@ ▶     private static double findAverage(String[] input) {  
        double result = 0;  
        for (String s : input) {  
            result += Integer.parseInt(s);  
        }  
        return result;  
    }  
}
```

Figure 122: Set breakpoints

Run the program in debug mode
Now start the program in debug mode.

From the main menu, select Run | Edit Configurations.
Enter arguments in the Program arguments field.

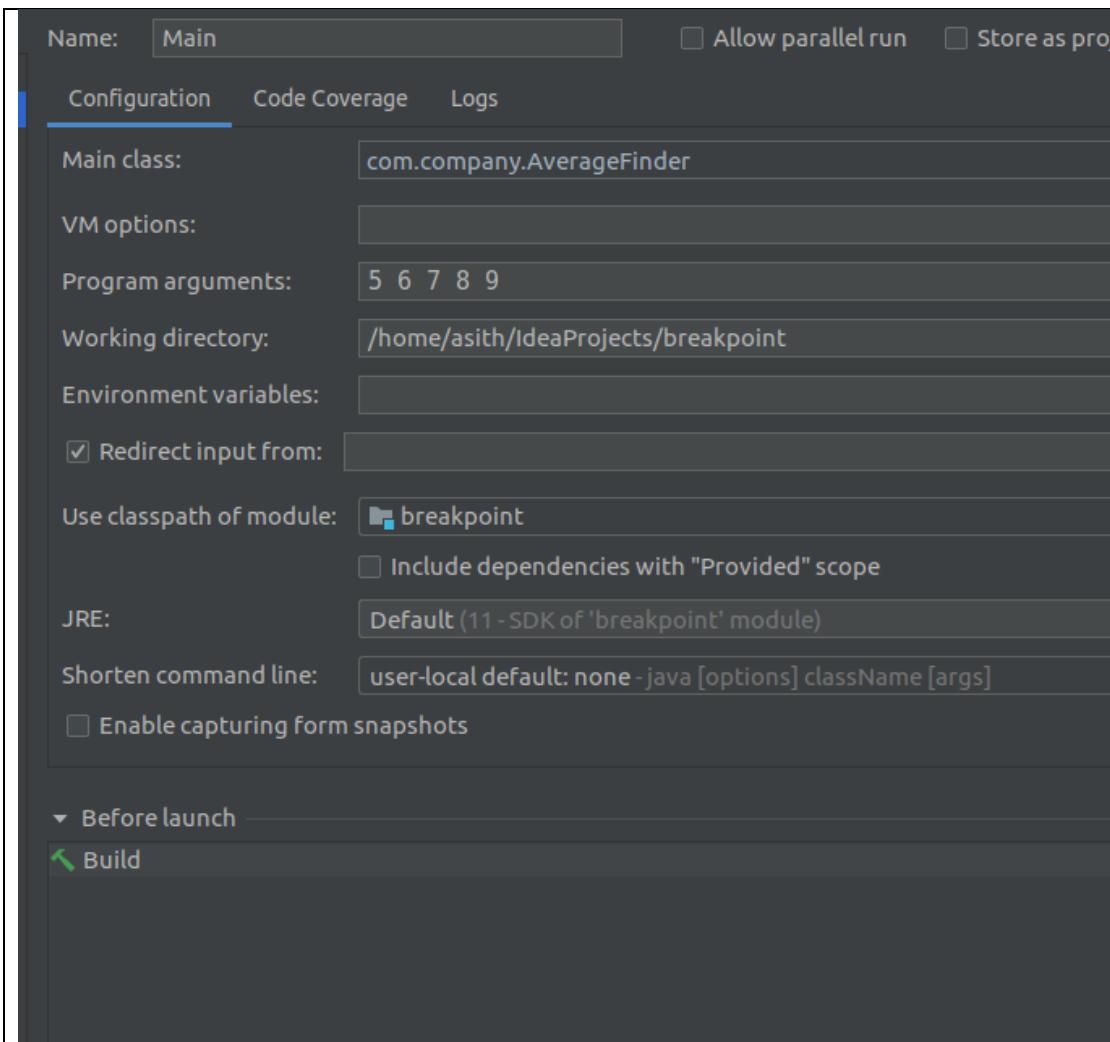


Figure 123:Program Arguments

Step through the program

```

args: {"5", "6", "7", "8", "9"}
double result = 0; result: 0.0
for (String s : input) { s: "5" input: {"5", "6", "7", "8",
    result += Integer.parseInt(s); result: 0.0 s: "5"
}
return result/input.length;
}

```

Figure 124:arguments apply to the program

Right now, the variable `s` contains the value "5". It is going to be converted to an Integer and be added

to `result`, which currently has the value of 5.0. No errors so far.

```
double result = 0; result: 5.0
for (String s : input) { s: "6" input: {"5", "6", "7", "8",
    result += Integer.parseInt(s); result: 5.0 s: "6"
}
return result/input.length;
```

Figure 125::arguments apply to the program2

Right now, the variable s contains the value "5". It is going to be converted to an Integer and be added to result, which currently has the value of 11.0. No errors so far.

```
double result = 0; result: 18.0
for (String s : input) { s: "8" input: {"5", "6", "7", "8", "9"}
    result += Integer.parseInt(s); result: 18.0 s: "8"
}
return result/input.length;
```

Figure 126::arguments apply to the program3

Right now, the variable s contains the value "5". It is going to be converted to an Integer and be added to result, which currently has the value of 11.0. No errors so far.

```
double result = 0; result: 35.0
for (String s : input) {
    result += Integer.parseInt(s);
}
double ff = result / input.length; ff: 7.0 result: 35.0 input: {"5", "6", "7", "8"
return ff; ff: 7.0
}
```

Figure 127::arguments apply to the program4

Two more steps take us to the return statement

Within Java OOP understanding, following topics were covered.

Learn how to use lambda expressions for programming.

Learn how to implement abstract methods of interfaces inside another class using lambda expression.

Revision of flow management (switch, for, if-else)

Learn how to use lambda expressions with the Stream API.

Learn how to use lambda expressions with no parameters and parameters.

Understanding how tow buffer and stream works with IO/NIO.

Learn relation between channels Selectors.

Get a good experience of testing web applications in Japanese language.

JSP understanding

I will add myself note here that related to JSP

Reference 1:

<https://beginnersbook.com/2013/05/servlet-tutorial/>

Reference 2:

Before going through the Jsp, I will discuss Servlets and CGI first. Because it is the base of JSP.

Servlets

Servlet (beginnersbook.com, n.d.) is a java program that runs inside JVM on the web server. It is used for developing dynamic web applications.

Let's understand what a dynamic **web application is?** A web application can be described as collection of web pages (e.g. a website) and when we call it dynamic, it simply means that the web pages are not same for all the users, web pages would be generated on server side based on the request made by client(user's browser)

The main **difference between static and dynamic web pages** is that a static page as the name suggests remains the same for all users, however a dynamic web page changes based on the request from the client (user's browser).

What a servlet does can be done by CGI (Common Gateway Interface), well its true but here is the thing – **CGI has several limitations such as performance, scalability, reusability etc. that a servlet doesn't have.**

Servlet is better than CGI.

Limitations of CGI

Server has to create a new CGI process (www3.ntu.edu.sg, n.d.) for every client request. For example, If 100 users are accessing the web application, then the server has to create 100 CGI processes to handle the request made by them.

How Servlet is better than CGI

CGI programs are handled by a new process every time a new request has been made. Unlike CGI, the servlet programs are handled by separate threads that can run concurrently more efficiently.

CGI programs can be written in any programming language that makes it mostly platform dependent as not all programming languages are platform independent. Servlet only uses Java as a programming language that makes it platform independent and portable. Another benefit of using java is that the servlet can take advantage of the object oriented programming features of java.

How Servlet Works

As I mentioned above that concurrent requests to the server are handled by threads, here is the graphical representation of the same –

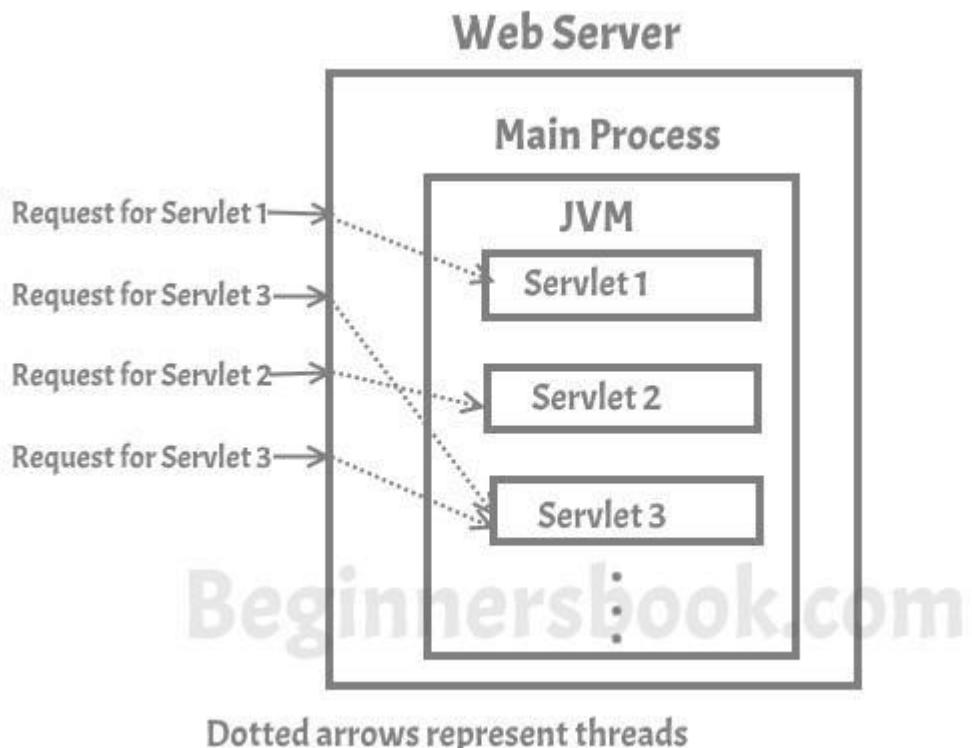


Figure 128: Feature of Servlet

Features of Servlet

1. Portable:

Servlet uses Java as a programming language. Since java is platform independent, the same holds true for servlets. For example, you can create a servlet on Windows operating system that uses GlassFish as a web server and later run it on any other operating system like Unix, Linux with Apache tomcat web server.

2. Efficient and scalable:

Once a servlet is deployed and loaded on a web server, it can instantly start fulfilling requests of clients. The web server invokes servlet using a lightweight thread so multiple client requests can be fulfilled by servlet at the same time using the multithreading feature of Java. Compared to CGI where the server has to initiate a new process for every client request, the servlet is truly efficient and scalable.

Robust:

By inheriting the top features of Java (such as Garbage collection, Exception handling, Java Security Manager etc.) the servlet is less prone to memory management issues and memory leaks. This makes development of web applications secure and less error prone.

Servlet API

You need to use the Servlet API to create servlets. There are two packages that you must remember while using API,

javax.servlet package that contains the classes to support generic servlet (protocol-independent servlet)

javax.servlet.http package that contains classes to support http servlet.

Hierarchy of packages

```
java.lang.Object
    |_extended by javax.servlet.GenericServlet
        |_extended by javax.servlet.http.HttpServlet
```

Figure 129: Hierarchy of packages

Every Servlet must implement the `java.servlet.Servlet` interface, you can do it by extending one of the following two classes

javax.servlet.GenericServlet

javax.servlet.http.HttpServlet

Generic Servlet

if you are creating a Generic Servlet then you must extend javax.servlet.GenericServlet class. GenericServlet class has an abstract service() method. Which means the subclass of GenericServlet should always override the service() method.

Signature of service() method:

```
public abstract void service(ServletRequest request, ServletResponse response)
    throws ServletException, java.io.IOException
```

Figure 130:Signature of service() method:

The service() method accepts two arguments:

ServletRequest object

ServletResponse object.

The request object tells the servlet about the request made by client while the response object is used to return a response back to the client

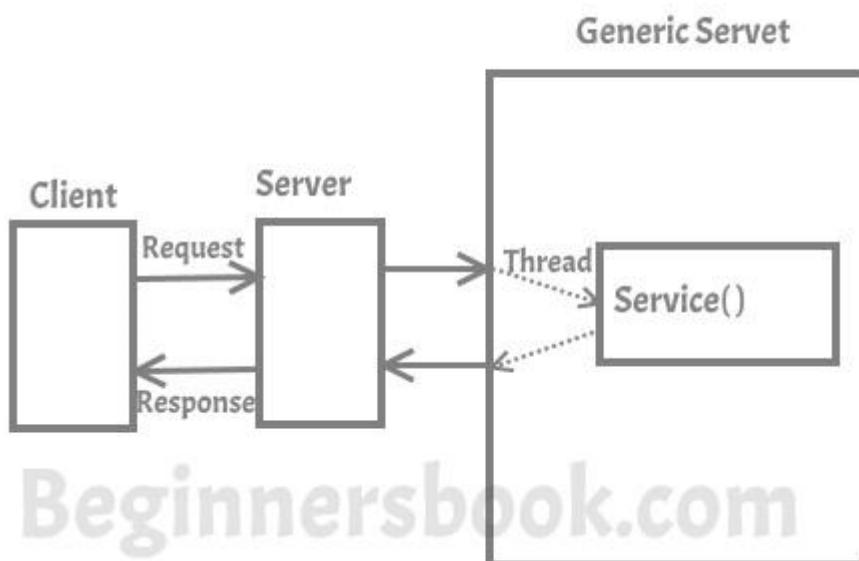


Figure 131:HTTP Servlet

HTTP Servlet

If you create an Http Servlet you must extend javax.servlet.http.HttpServlet class, which is an abstract class. Unlike Generic Servlet, the HTTP Servlet doesn't override the service() method. Instead it overrides one or more of the following methods. It must override at least one method from the list below:

doGet() – This method is called by the servlet service method to handle the HTTP GET request from the client. The Get method is used for getting information from the server

doPost() – Used for posting information to the Server

doPut() – This method is similar to doPost method but unlike doPost method where we send information to the server, this method sends file to the server, this is similar to the FTP operation from client to server

doDelete() – allows a client to delete a document, webpage or information from the server

init() and destroy() – Used for managing resources that are held for the life of the servlet

getServletInfo() – Returns information about the servlet, such as author, version, and copyright.

always recommend you to use **HttpServlet** instead of the **GenericServlet**. HttpServlet is easier to work with, and has more methods to work with than GenericServlet.

What is JSP?

Java Server Pages (JSP) is a technology which is used to develop web pages by inserting java code into the HTML pages by making special JSP tags (JSP understanding, n.d.). The JSP tags which allow java code to be included into it are <% ----java code----%>.

It can be used as an HTML page, which can be used in forms and registration pages with the dynamic content into it.

We can share information across pages using request and response objects.

It is much better than Servlet and Common Gateway Interface (CGI).

Java server pages JSP are built over Java Servlets API. Hence, it has access to all Java APIs, even it has access to JNDI, JDBC EJB and other components of java.

JSP is used in MVC architecture (which will be covered in MVC architecture topic) as a view layer.

The request is processed by a view layer which is JSP and then a servlet layer which is a java servlet and then finally to a model layer class which interacts with the database.

JSP is an important part of Java EE, which is a platform for enterprise level applications

Advantages of JSP

The advantage of JSP is that the programming language used is JAVA, which is a dynamic language and easily portable to other operating systems.

It is very much convenient to modify the regular HTML. We can write the servlet code into the JSP.

Performance and scalability of JSP are very good because JSP allows embedding of dynamic elements in HTML pages.

As it is built on Java technology, hence it is platform independent and not depending on any operating systems.

Also, it includes the feature of multithreading of java into it.

We can also make use of exception handling of java into JSP.

It enables a separate presentation layer with the business logic layer in the web application.

What is JSP Life Cycle?

JSP Life Cycle is defined as translation of JSP Page into servlet.

JSP Page needs to be converted into a servlet first in order to process the service requests.

The Life Cycle starts with the creation of JSP and ends with the disintegration of that.

learn different phases of JSP Life Cycle

When the browser asks for a JSP, JSP engine first checks whether it needs to compile the page. If the JSP is last compiled or the recent modification is done in JSP, then the JSP engine compiles the page.

Compilation process of JSP page involves three steps:

Parsing of JSP

Turning JSP into servlet

Compiling the servlet

JSP Lifecycle is depicted in the below diagram.

Following steps explain the JSP life cycle:

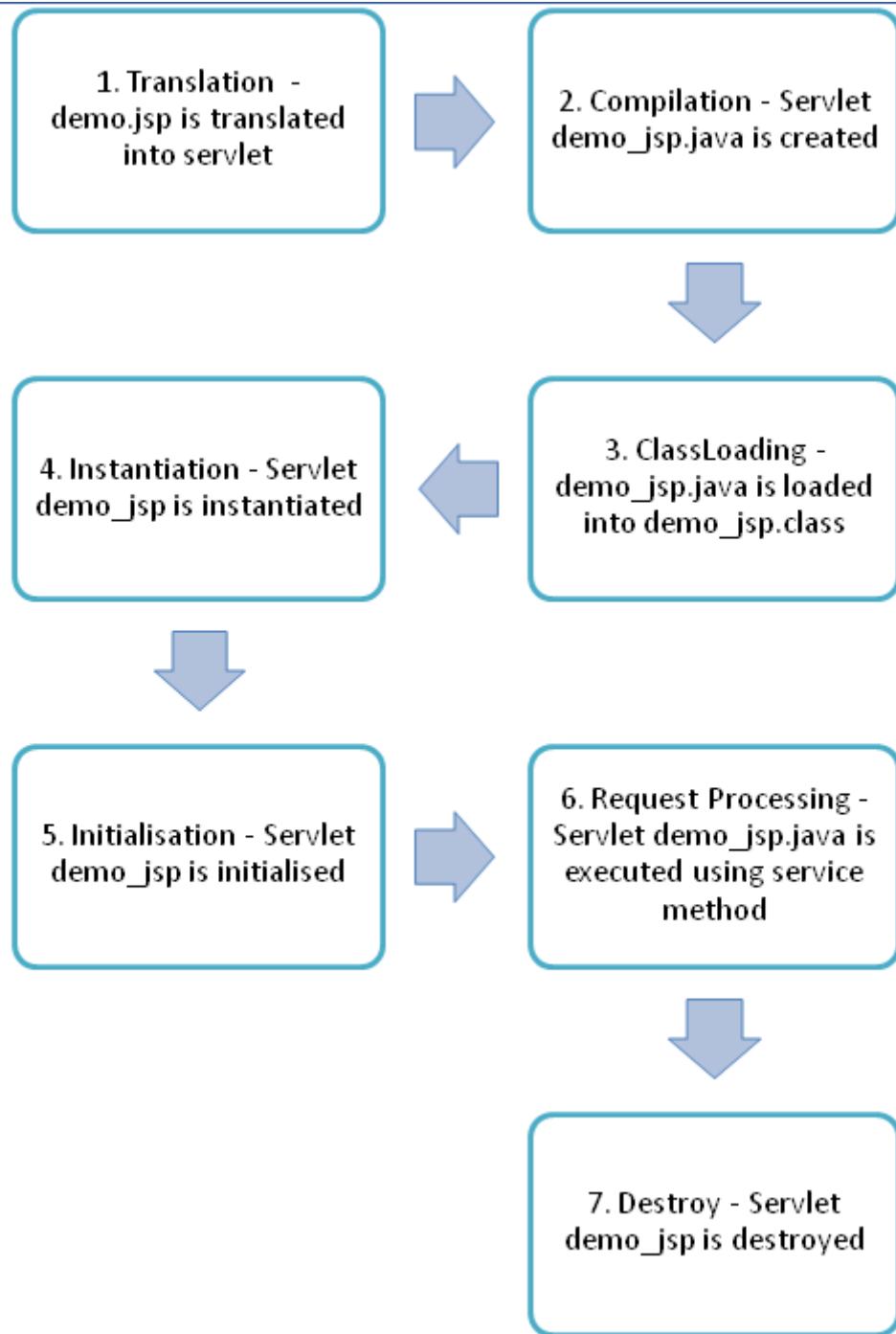


Figure 132:the JSP life cycle

1. Translation of the JSP Page:

A Java servlet file is generated from a JSP source file. This is the first step of JSP life cycle. In the translation phase, the container validates the syntactic correctness of JSP page and tag files.

In the above pictorial description, **Example.jsp** is translated to **Example_jsp.java** in the first step

```
<%@ page contentType="text/html;charset=UTF-8" language="java" %>
<!DOCTYPE html>
<html>
<head>
    <title>Demo JSP</title>
</head>
<body>
<%
    int demvar=0;
%>
Count is:<%= demvar%>
this is asith!!!!1
</body>
</html>
```

Figure 133:Example_jsp.java

Output:

Count is:0 this is asith!!!!1

Figure 134:Example_jsp.java output

When a JSP is first accessed, Tomcat converts the JSP into a servlet;

compile the servlet, and execute the servlet.

you shall understand that servlet is "HTML inside Java", whereas JSP is "Java inside HTML".

Revisit Java Servlets

A typical Java servlet (as shown below) contains three kinds of methods:

init()

destroy()

and one or more service() methods such as

doGet()

doPost()

init() runs when the servlet is loaded.

destroy() runs when the servlet is unloaded.

service() runs once per HTTP request.

The service() method takes two arguments: request and response.

corresponding to the HTTP request and response messages respectively. A PrintWriter called out is created for writing out the response to the network.

JSP Scripting Elements

JSP provides the following scripting elements:

- JSP Comment <%-- comments -->
- JSP Expression <%= Java Expression %>
- JSP Scriptlet <% Java Statement(s) %>
- JSP Directive <%@ page|include ... %>

To simplify the access of the HTTP *request* and *response* messages, JSP has *pre-defined* the following variables:

- request: corresponds to the HTTP request message.
- response: corresponds to the HTTP response message.
- out: corresponds to the HTTP response message's output stream.

JSP comment <%-- comments --%>

JSP comments <%-- comments --%> are ignored by the JSP engine. For example,

<%-- anything but a closing tag here will be ignored --%>

JSP Expression <%= Java Expression %>

JSP Expression can be used to insert a *single* Java expression directly into the response message. This expression will be placed inside a out.print() method. Hence, the

expression will be evaluated and printed out as part of the response message. Any valid Java expression can be used. There is no semicolon at the end of the expression.

For examples:

```
<p>The square root of 5 is <%= Math.sqrt(5) %></p>
<h5><%= item[10] %></h5>
<p>Current time is: <%= new java.util.Date() %></p>
```

You can use the pre-defined variables, such as request, in the expressions. For examples:

```
<p>You have choose author <%= request.getParameter("author") %></p>
<%= request.getRequestURI() %>
<%= request.getHeader("Host") %>
```

JSP Scriptlet <% Java statement(s) %>

JSP scriptlets allow you to do more *complex operations* than inserting a single Java expression (with the JSP expression).

```
<%
String author = request.getParameter("author");
if (author != null && !author.equals("")) {
%>
<p>You have choose author <%= author %></p>
<%
}
%>
```

Figure 135:complex operations JSP

JSP Directive <% @ page|include ... %>

JSP directives provide instructions to the JSP engine. The syntax of the JSP directive is:

```
<%@ directive_name  
attribute1="value1"  
attribute2="value2"  
.....  
attributeN="valueN" %>
```

Figure 136: JSP directives

JSP page Directive

The "page" directive lets you import classes and customize the page properties. For examples,

JSP include Directive

The "include" directive lets you include another file(s) at the time when the JSP page is compiled into a servlet. You can include any JSP files, or static HTML files. You can use include directive to include navigation bar, copyright statement, logo, etc. on every JSP pages. The syntax is:

```
<%@ include file="header.html" %>  
.....  
<%@ include file="footer.html" %>
```

Figure 137: JSP include Directive

JSP Declaration

<%! ----- Use to declare----- %>

After self study about the JSP, the supervisor asked us to develop a sample bank application using JSP which doesn't have database connectivity.

This is the Bank system that was developed using **JSP, servlet** along with MVC architecture.

1. **.jsp used as view**
2. **.java (servlet) used as controllers.**

This is the index page and it consists of one button called **login**.

Every user has to press this button and when press the button it will redirect to login form page,

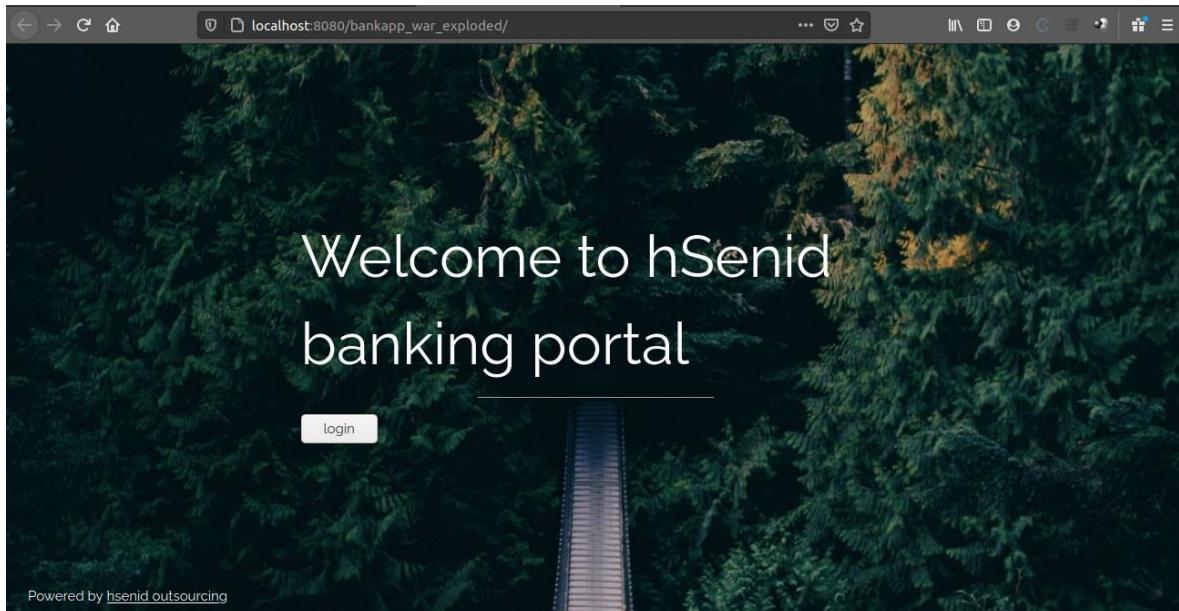


Figure 138:Servlet bank app login

In this step, User has to enter his/her **name**, **bank account number** and **bank balance** for the continuation.

Username	asith ishantha madurawala
Help	AccountNumber
1432456476567	
BankAmount	350000
Login	

Figure 139:bank app login info

After entering required details, the system will redirect to another web page and it shows all the information that was entered by the user.

Users can do the confirmation of his/her entered information.

This window provides two functionalities to the user.

Form validation

When the username field is empty, it prompts an error message called "username must be filled out".

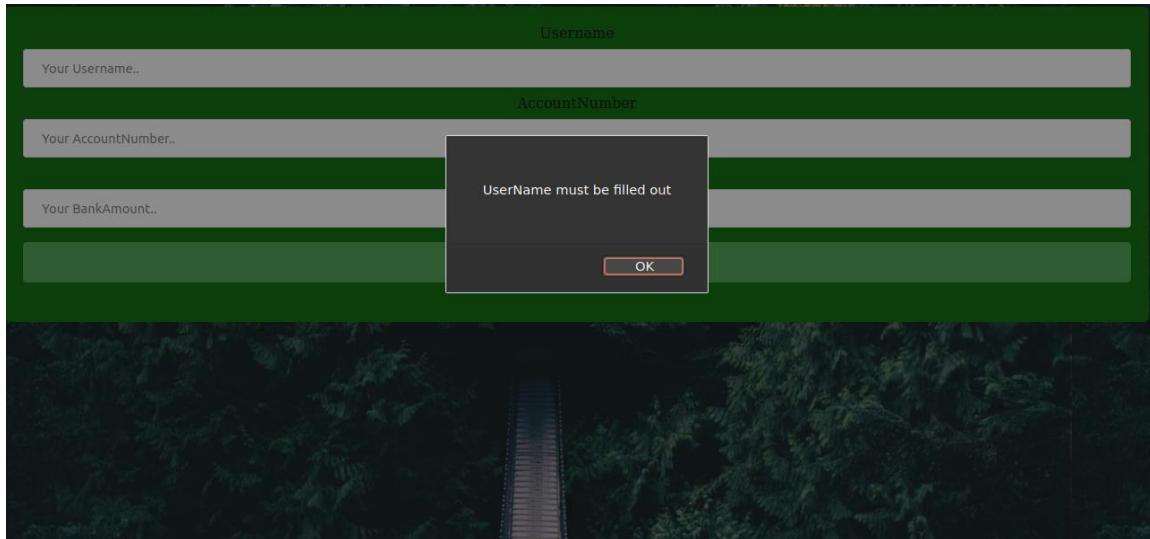


Figure 140:bank app form validation

When the account number field is empty, it prompts an error message called "account number must be filled out".

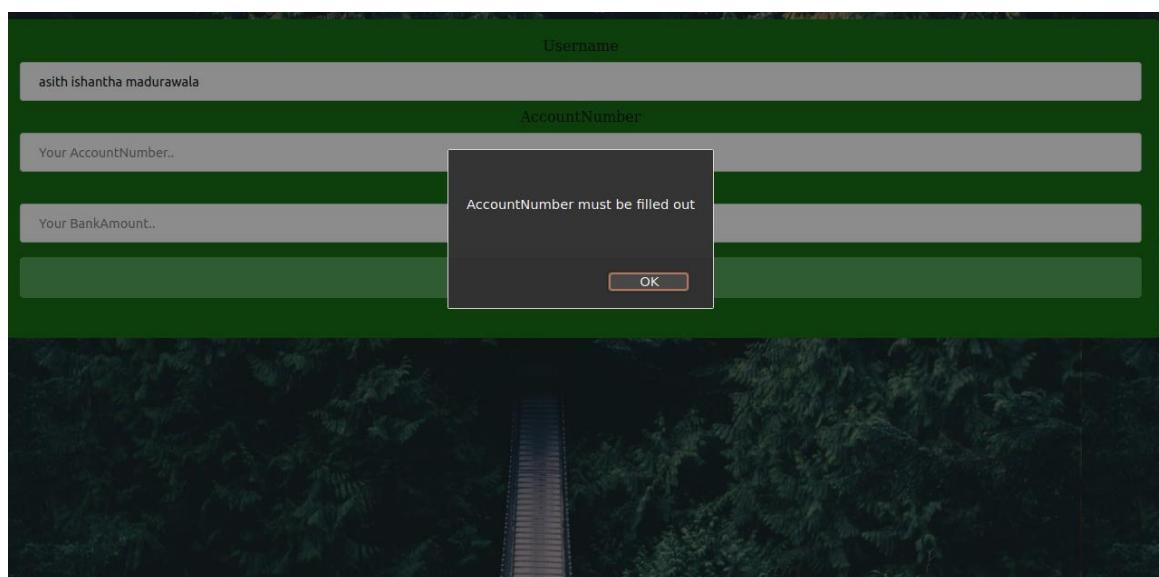


Figure 141:account number must be filled out".

When the bank amount field is empty, it prompts an error message called "bank amount must be filled out".

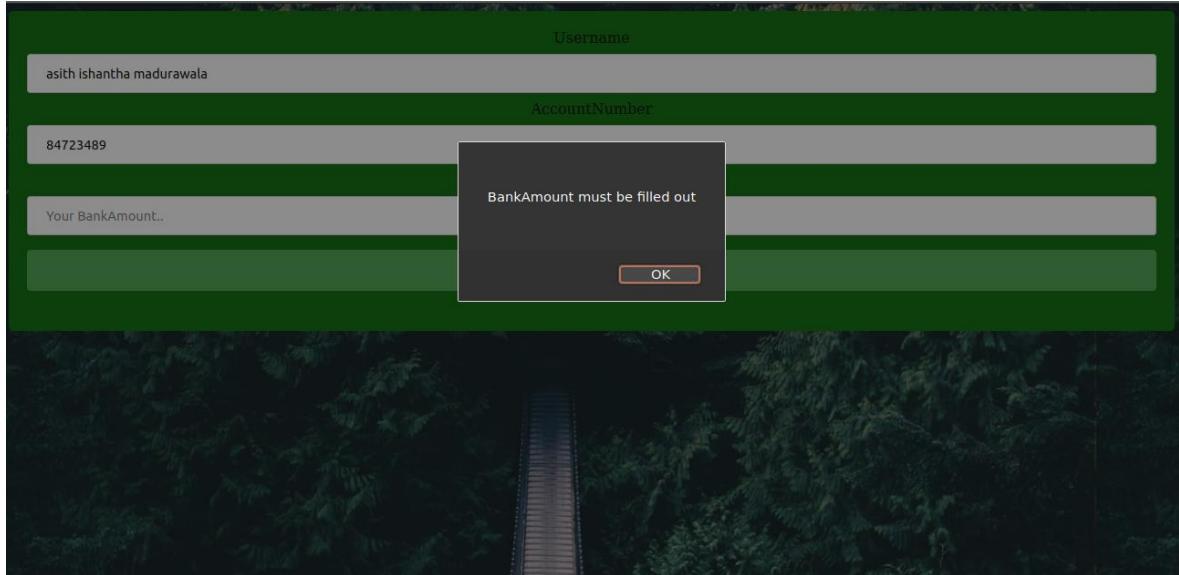


Figure 142: bank amount must be filled out”.

After successfully login with valid credentials, you can see the entered details and two facilities provided by the system.

- 1. Withdraw money**
- 2. Deposit money**

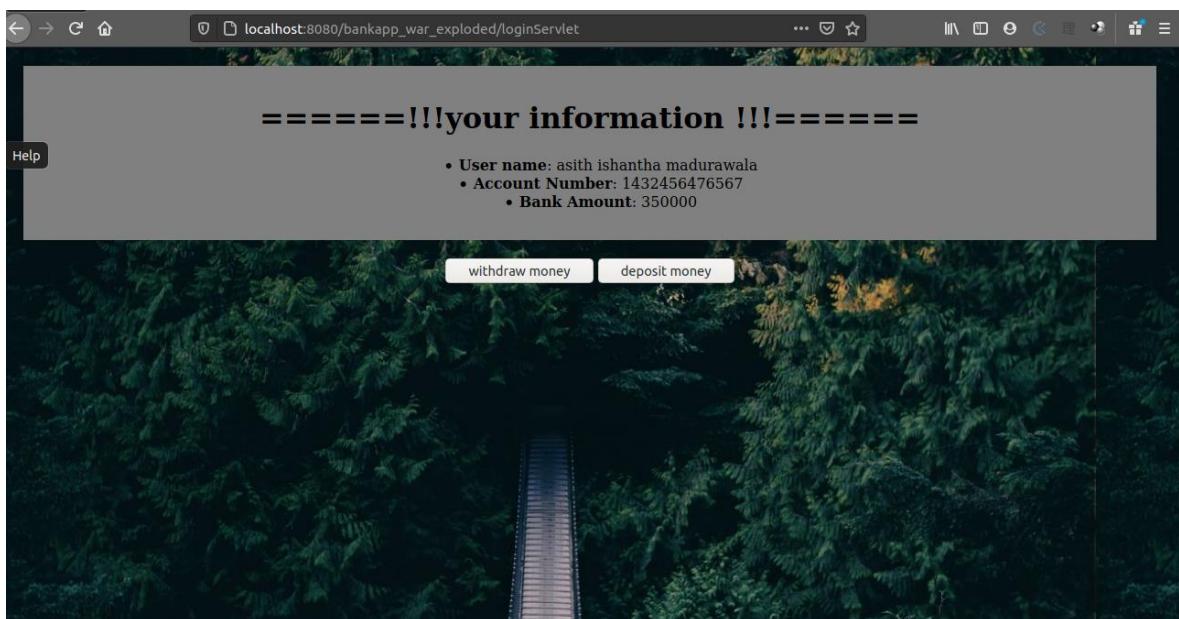


Figure 143: withdraw money and deposit money

If a user selects a withdrawal option, he will be able to withdraw money from the system. User has to enter a bank account again as a two step verification and the amount that they wish to withdraw.

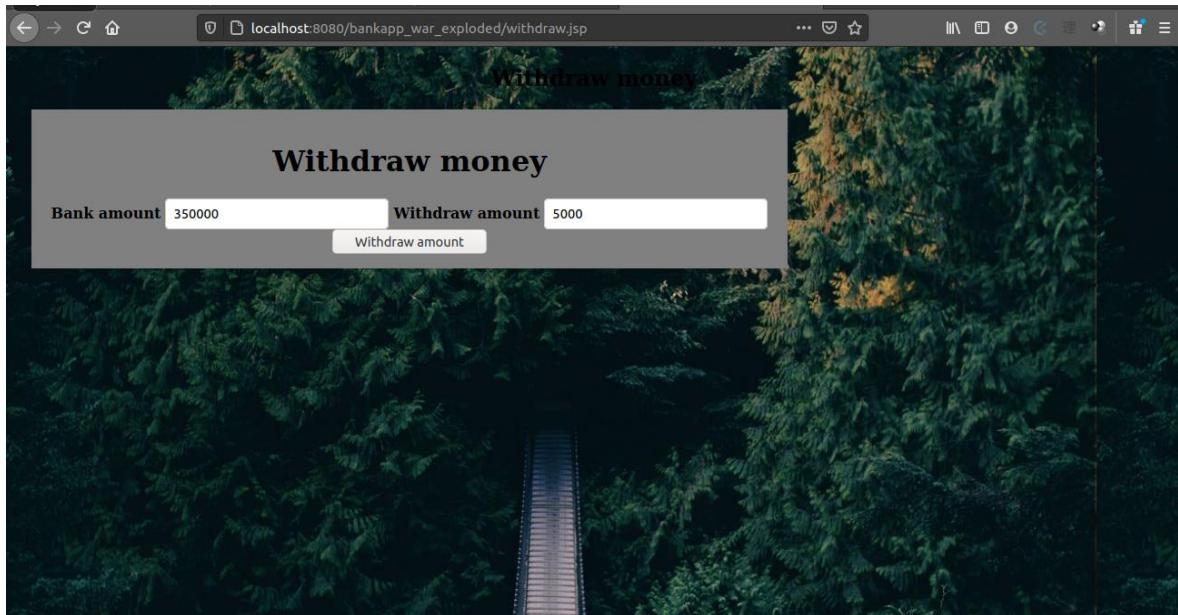


Figure 144: withdraw money enter amount

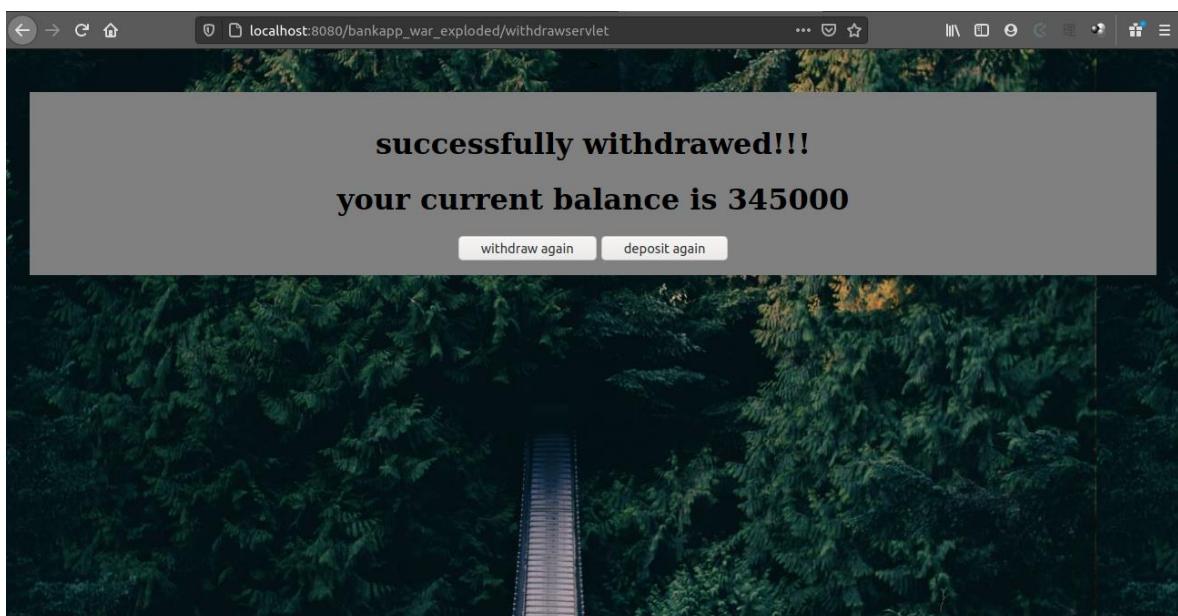


Figure 145: successfully withdrawal

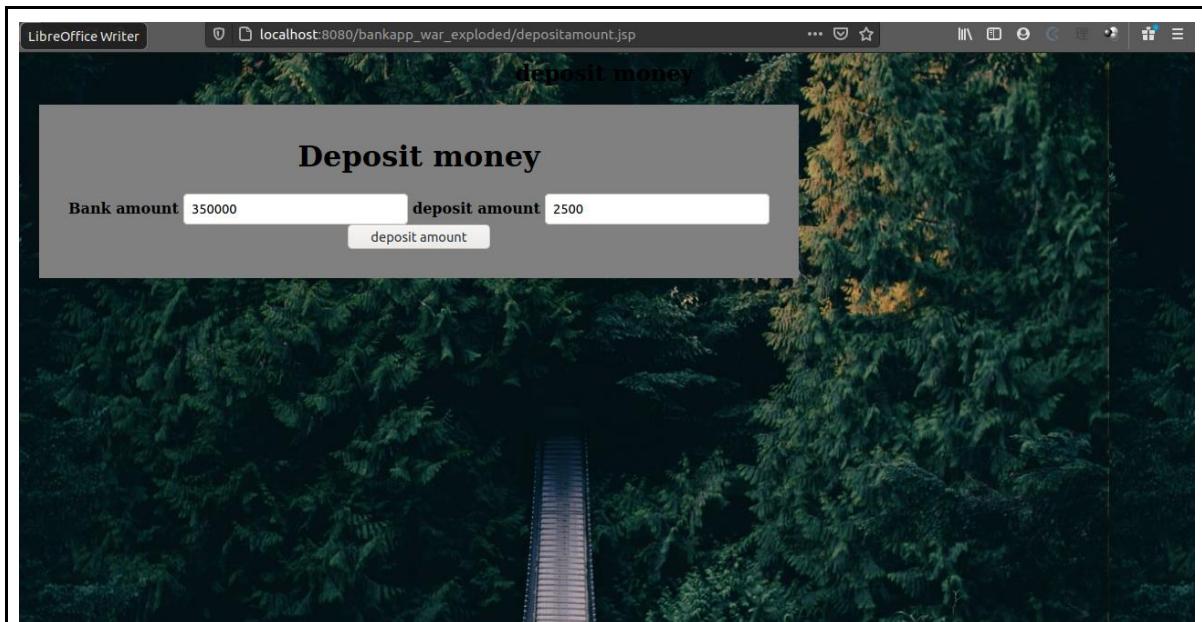


Figure 146:bank app deposit money

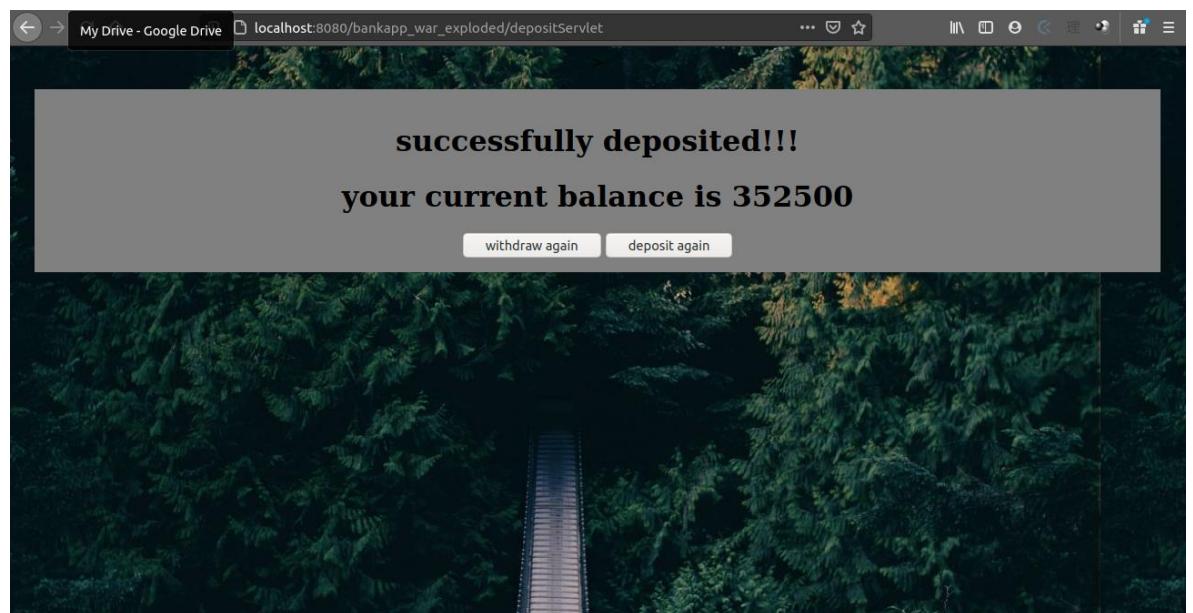


Figure 147:successfully deposited

If we try to withdraw an amount, that is greater than the account balance. System is prompt an error message called “insufficient account balance”

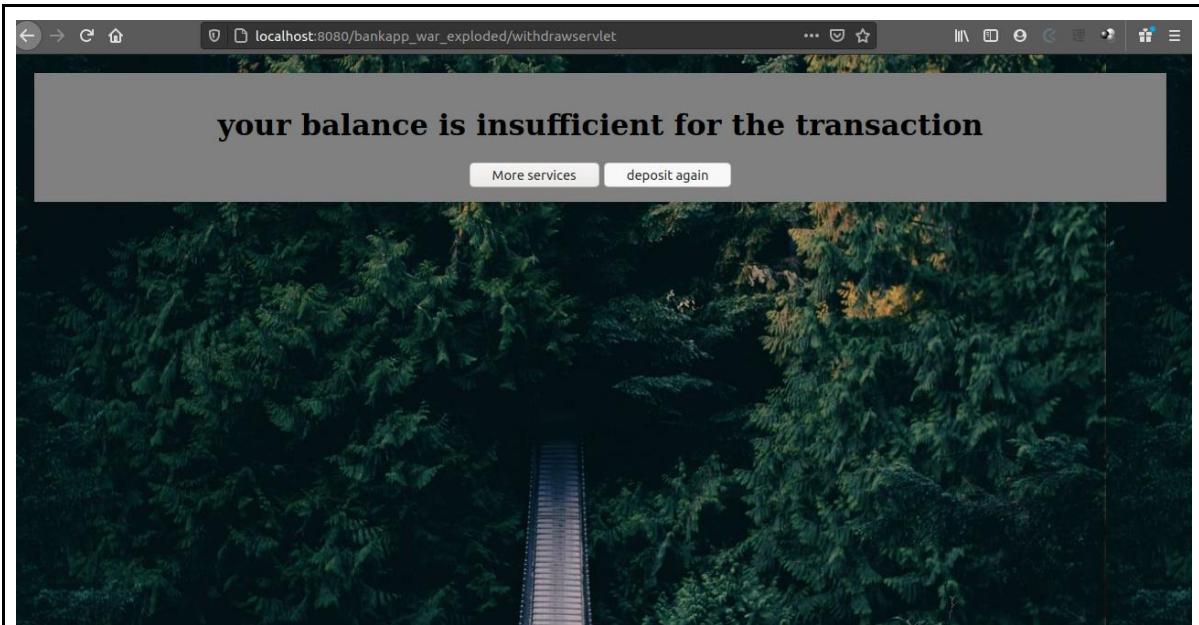


Figure 148:Insufficient balance

Maven

First of all, I will add my self-study note regarding Maven here.

Installation

Maven is a Java tool, so you must have JAVA installed in order to proceed.

Ensure JAVA_HOME environment variable is set and points to your JDK installation

Extract distribution archive in any directory.

It allows the developer to create projects using Project Object Model and plugins.

```
tar xzvf apache-maven-3.6.3-bin.tar.gz
```

2. Setup environment variables

need to set up the environment variables. To do so, open your **text editor** and create a new file named mavenenv.sh inside of the /etc/profile.d/ directory.

```
sudo nano /etc/profile.d/maven.sh
```

```
export JAVA_HOME=/home/asith/softwares/jdk-11.0.3
export M2_HOME=/opt/maven
export MAVEN_HOME=/opt/maven
export PATH=${M2_HOME}/bin:${PATH}
```

Make the script executable with chmod :

```
sudo chmod +x /etc/profile.d/maven.sh
```

Finally load the environment variables using the source command:

```
source /etc/profile.d/maven.sh
```

Verify the installation

```
mvn -version
```

```
asith@hsentid-Vostro-3650:/opt/tomcat/apache-tomcat-7.0.105/webapps$ mvn -version
Apache Maven 3.6.3 (cecedd343002696d0abb50b32b541b8a6ba2883f)
Maven home: /home/asith/softwares/maven
Java version: 11.0.3, vendor: Oracle Corporation, runtime: /home/asith/softwares/jdk-11.0.3
Default locale: en_US, platform encoding: UTF-8
OS name: "linux", version: "5.4.0-45-generic", arch: "amd64", family: "unix"
```

Figure 149:mvn -version

Maven Overview - Core Concepts

Maven is centered around the concept of POM files (Project Object Model). A POM file is an XML representation of project resources like source code, test code, dependencies (external JARs used) etc. The POM contains references to all of these resources. The POM file should be located in the root directory of the project it belongs to.

Here is a diagram illustrating how Maven uses the POM file, and what the POM file primarily contains:

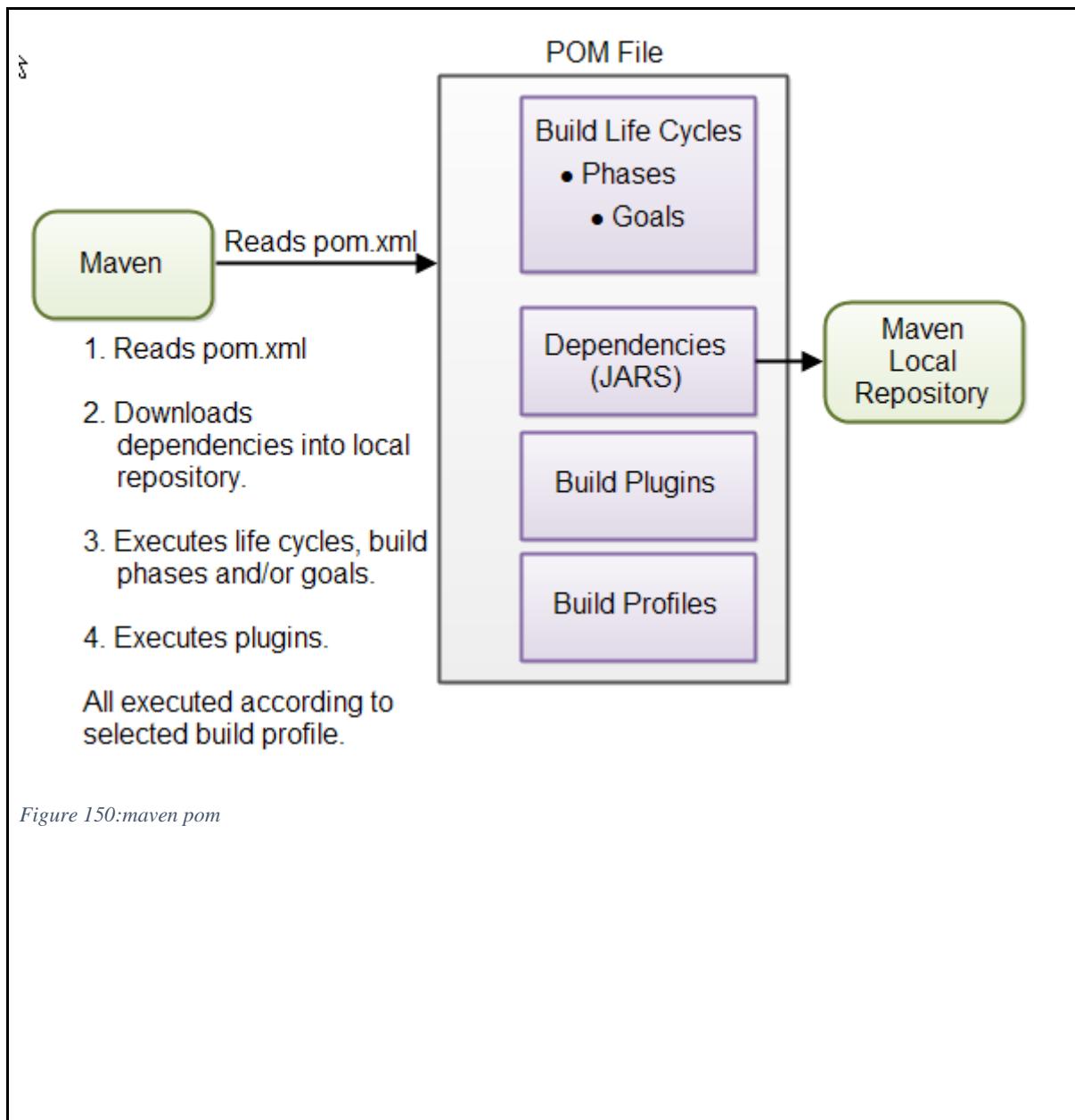


Figure 150:maven pom

Software Unit Testing

In web bank applications, there are logic tests to check whether that logic is working properly or not. Many test classes were written to check these conditions. And run those test cases.

There are the methods I was going to test.

Deposit
Withdraw

```

    public double deposit(double intamount, double intdeposited) {
        double balance= intamount + intdeposited;

        return balance;
    }

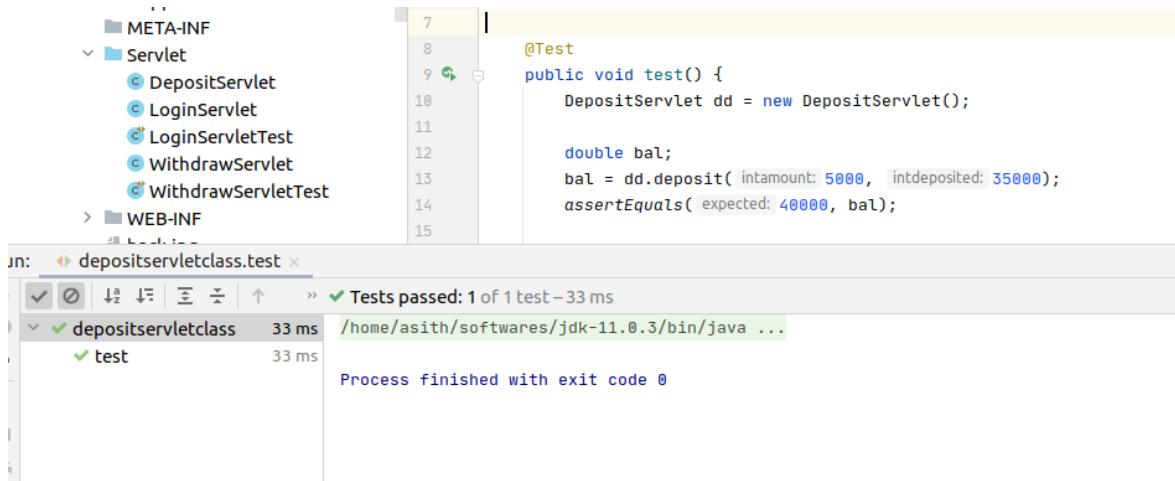
    public double withdraw(double intamount, double intwithdrawed) {
        double balance= intamount - intwithdrawed;

        return balance;
    }

```

This is the test method I have written to test depositAmount logic.
That test case successfully passed and the expected result was same with actual result.

Deposit servlet class- (test class / test method)



The screenshot shows an IDE interface with two main panes. On the left is a file tree showing a project structure with META-INF, Servlet (containing DepositServlet, LoginServlet, LoginServletTest, WithdrawServlet, WithdrawServletTest), and WEB-INF. On the right is a code editor with the following Java code:

```

    public class DepositServlet {
        @Test
        public void test() {
            DepositServlet dd = new DepositServlet();
            double bal;
            bal = dd.deposit( intamount: 5000, intdeposited: 35000 );
            assertEquals( expected: 40000, bal );
        }
    }

```

Below the code editor is a terminal window showing the test results:

```

    Tests passed: 1 of 1 test - 33 ms
    depositServletclass 33 ms /home/asith/softwares/jdk-11.0.3/bin/java ...
    test 33 ms
    Process finished with exit code 0

```

Figure 151: Deposit servlet class- (test class / test method)

withdraw servlet class- (test class / test method)

The screenshot shows an IDE interface with Java code in the editor and a terminal window below it.

```

1 import org.junit.Test;
2
3 import static org.junit.jupiter.api.Assertions.assertEquals;
4
5 public class withdrawservletclass {
6
7     @Test
8     public void test() {
9         WithdrawServlet dd = new WithdrawServlet();
10
11
12         double bal;
13         bal = dd.withdraw( intamount: 35000, intwithdrawed: 5000 );
14         assertEquals( expected: 30000, bal );
15     }
16 }
17
18 }
```

Run: withdrawservletclass.test

Tests passed: 1 of 1 test – 52 ms

withdrawservletclass 52 ms /home/asith/softwares/jdk-11.0.3/bin/java ...
test 52 ms
Process finished with exit code 0

Figure 152:withdraw servlet class- (test class / test method)

Note: I used Junit to check the logic (Unit Testing Junit, n.d.)

Java FX

JavaFX is a Java library used to build Rich Internet Applications (Java FX, n.d.). The applications written using this library can run consistently across multiple platforms. The applications developed using JavaFX can run on various devices such as Desktop Computers, Mobile Phones, TVs, Tablets, etc.

To develop **GUI Applications** using Java programming language, the programmers rely on libraries such as **Advanced Windowing Toolkit** and **Swing**. After the advent of JavaFX, these Java programmers can now develop GUI applications effectively with rich content.

To develop **Client Side Applications** with rich features, the programmers used to depend on various libraries to add features such as Media, UI controls, Web, 2D and 3D, etc. JavaFX includes all these features in a single library. In addition to these, the developers can also access the existing features of a Java library such as **Swing**.

Features of JavaFX

Following are some of the important features of JavaFX –

Java Library

It is a Java library which consists of many classes and interfaces that are written in Java.

FXML

FXML is the XML based Declarative markup language. The coding can be done in FXML to provide the more enhanced GUI to the user.

Scene Builder

Scene Builder generates FXML mark-up which can be ported to an IDE.

Web view

Web pages can be embedded with JavaFX applications. Web View uses WebKitHTML technology to embed web pages.

Built in UI controls

JavaFX contains Built-in components which are not dependent on the operating system. The UI components are just enough to develop a full featured application.

CSS like styling

JavaFX code can be embedded with the CSS to improve the style of the application. We can enhance the view of our application with the simple knowledge of CSS.

Swing interoperability

The JavaFX applications can be embedded with swing code using the Swing Node class. We can update the existing swing application with the powerful features of JavaFX.

Canvas API

Canvas API can provide the methods for drawing directly in an area of a JavaFX scene.

Rich Set of APIs

JavaFX provides a rich set of APIs to develop GUI applications.

Integrated Graphics Library

An integrated set of classes are provided to deal with 2D and 3D graphics.

Graphics Pipeline

JavaFX graphics are based on Graphics rendered pipeline(prism). It offers smooth graphics which are hardware accelerated.

High Performance Media Engine

The media pipeline supports the playback of web multimedia on a low latency. It is based on a Gstreamer Multimedia framework.

Self-contained application deployment model

Self Contained application packages have all of the application resources and a private copy of Java and JavaFX Runtime.

JavaFX simple application

JavaFX is a software platform that is used for developing desktop applications that can run across a wide variety of devices.

Download the JavaFX SDK

Using this link, we can download the [JavaFX SDK](#) package suitable for your operating system.

Unpack the archive and place the folder to a meaningful location, for example:

`/home/asith/softwares/openjfx-11.0.2_linux-x64_bin-sdk/javafx-sdk-11.0.2`

Create a new project

Launch IntelliJ IDEA.

If the Welcome screen opens, click Create New Project.

Otherwise, from the main menu, select File | New | Project.

In the New Project wizard, select JavaFX from the list on the left.

From the Project SDK list, select the [JDK](#) that you want to use in your project.

If the JDK is installed on your computer, but not defined in the IDE, select Add JDK and specify the path to the JDK home directory.

If you don't have the necessary JDK on your computer, select Download JDK.

Add the JavaFX library

These actions are required if you use Java 11 and later. If you use Java 10 and earlier, [run the sample application](#) created together with the project to make sure that everything is working as intended.

from the main menu, select File | Project Structure Ctrl+Alt+Shift+S

Open the Libraries section, click +, and select Java.

Specify the path to the lib folder in the JavaFX SDK package, for example:
/home/asith/softwares/openjfx-11.0.2_linux-x64_bin-sdk/javafx-sdk-11.0.2/lib

In the Choose Modules dialog, select the necessary module and click OK.

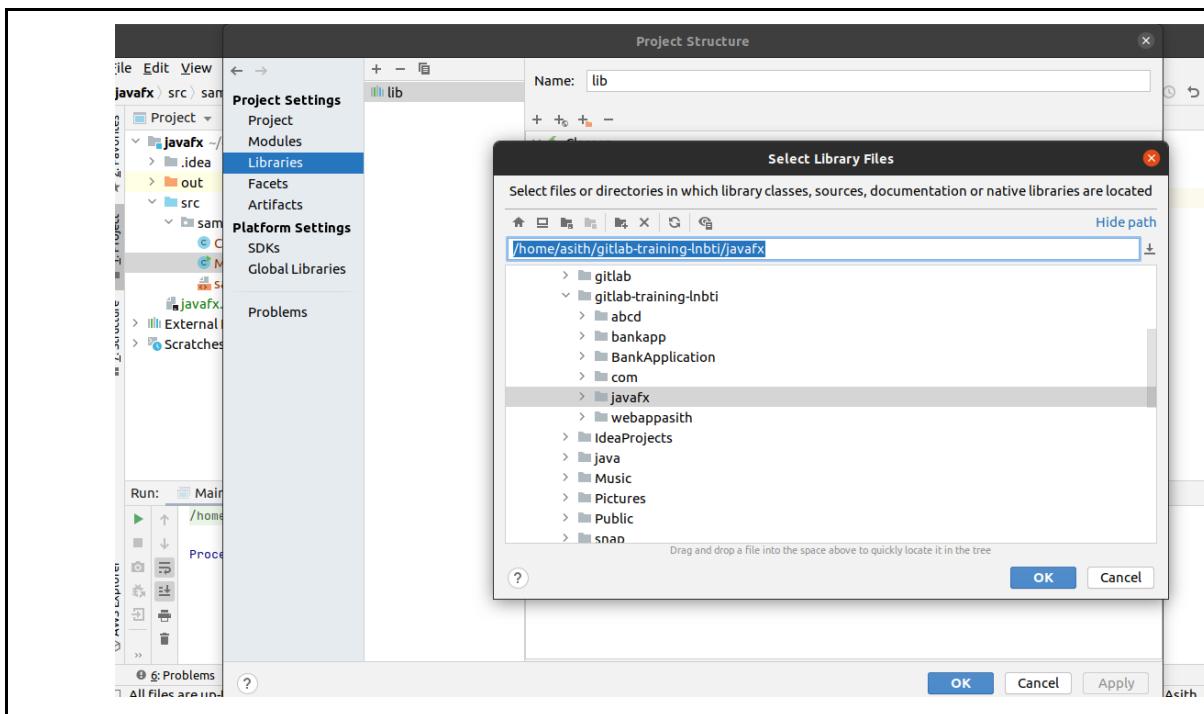


Figure 153:select the necessary module

Apply the changes and close the Project Structure dialog.

Add VM options

These actions are required if you use Java 11 and later. If you use Java 10 and earlier, you can [run the sample application](#) created together with the project to make sure that everything is working as intended.

From the main menu, select Run | Edit Configurations.

Select Application | Main from the list on the left.

In the VM options field, specify: --module-path %PATH_TO_FX% --add-modules javafx.controls,javafx.fxml

Instead of %PATH_TO_FX%, specify the path to the JavaFX SDK lib directory, for example: /home/asith/software/openjfx-11.0.2_linux-x64_bin-sdk/javafx-sdk-11.0.2/lib

If you're using JavaFX SDK 14 together with OpenJDK 14, add one more VM

option: **--add-exports javafx.graphics/com.sun.javafx.sg.prism=ALL-UNNAMED**

Apply the change and close the dialog.

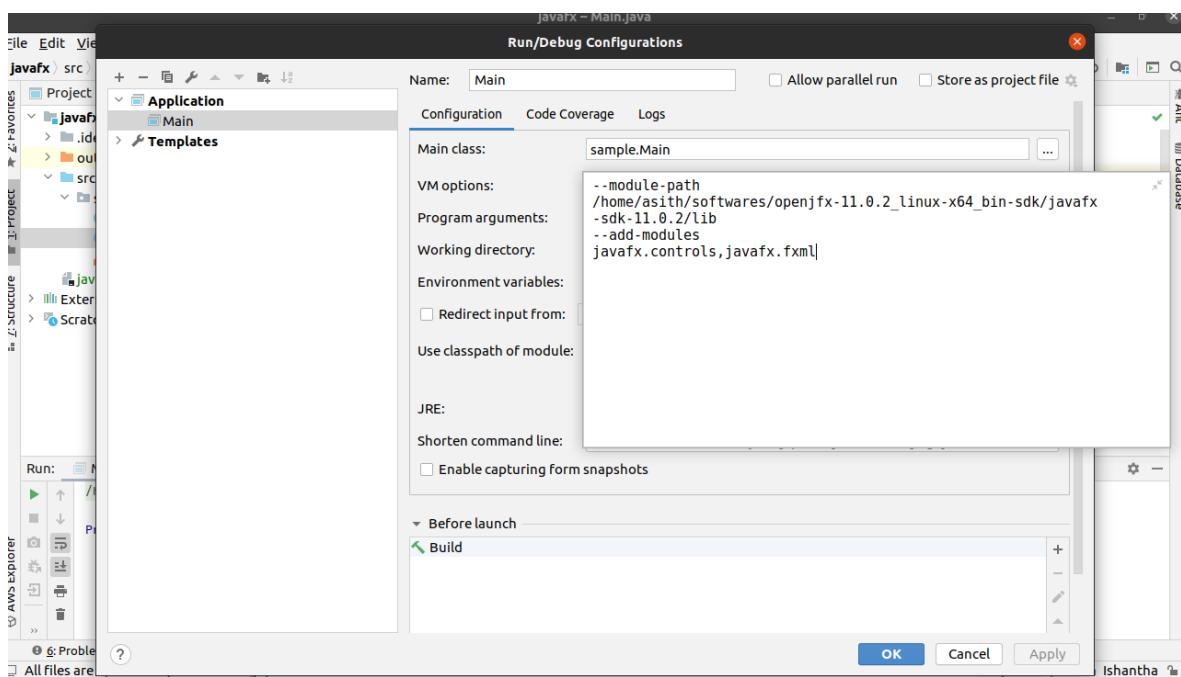


Figure 154:JavaFX SDK 14 together with OpenJDK 14, add one more VM option: **--add-exports javafx.graphics/com.sun.javafx.sg.prism=ALL-UNNAMED**

Run the application

From the main menu, select Run | Run 'Main' Shift+F10.

The IDE starts compiling your code.

When the compilation is complete, the application window appears.

This means that the project is configured correctly and everything works as it should.

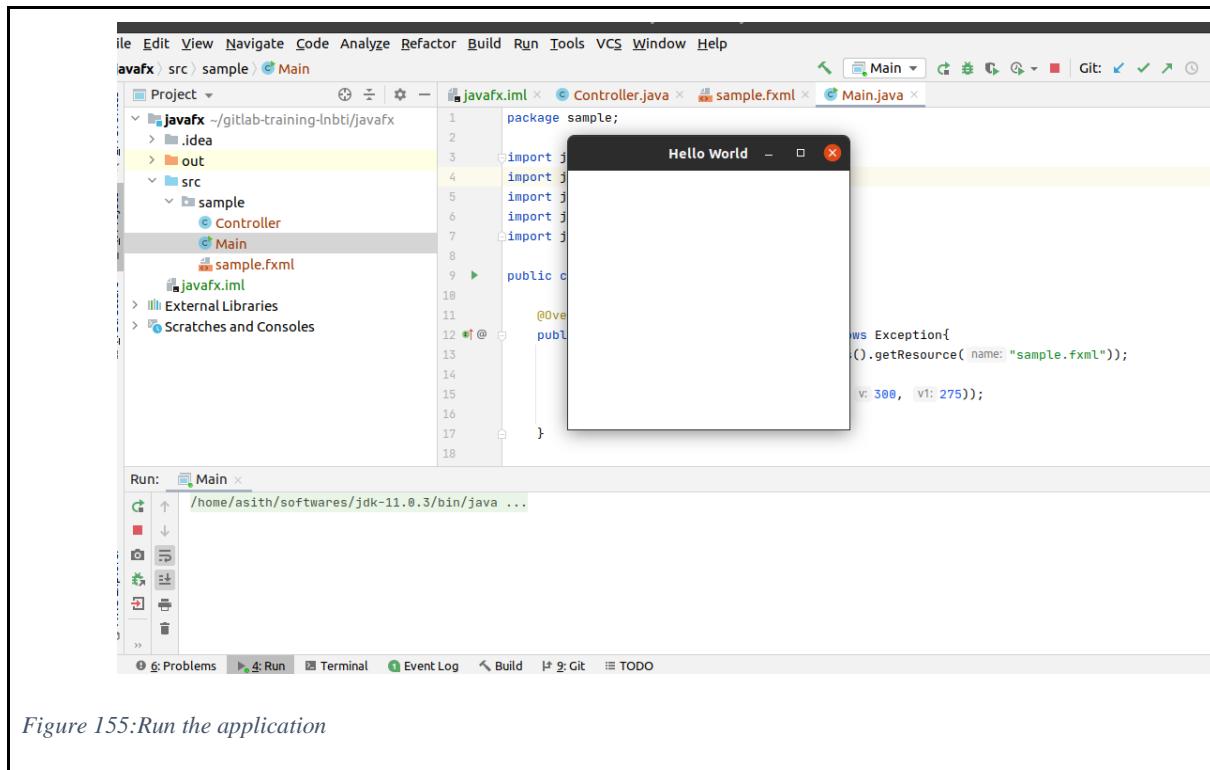


Figure 155:Run the application

Spring Web application

I will post my self-study note regarding Spring MVC here (Spring MVC tutorial, n.d.).

Spring has many good things, when comparing **spring MVC** against **Servlet MVC**. **Servlet mvc consists with** servlets and jsp. JSP acts as a view and servlets act as controllers. Spring consists of controllers and do the same thing in a very effective way.

In servlets, we have a HTTP Servlet and generic servlet class and we can use methods like service, doGet , doPost . And the client sends a request from the client side(jsp) and it will go to the server side(servlet) through web.xml and that is how it works. But if we have many controllers (servlet classes) it is difficult to manage in servlet MVC. and each servlet behaves like one controller. In spring we can have a multi action controller.

In spring to handle all controllers there is a one controller that is created by spring called a dispatcher servlet. All requests will go to the dispatcher through xml.

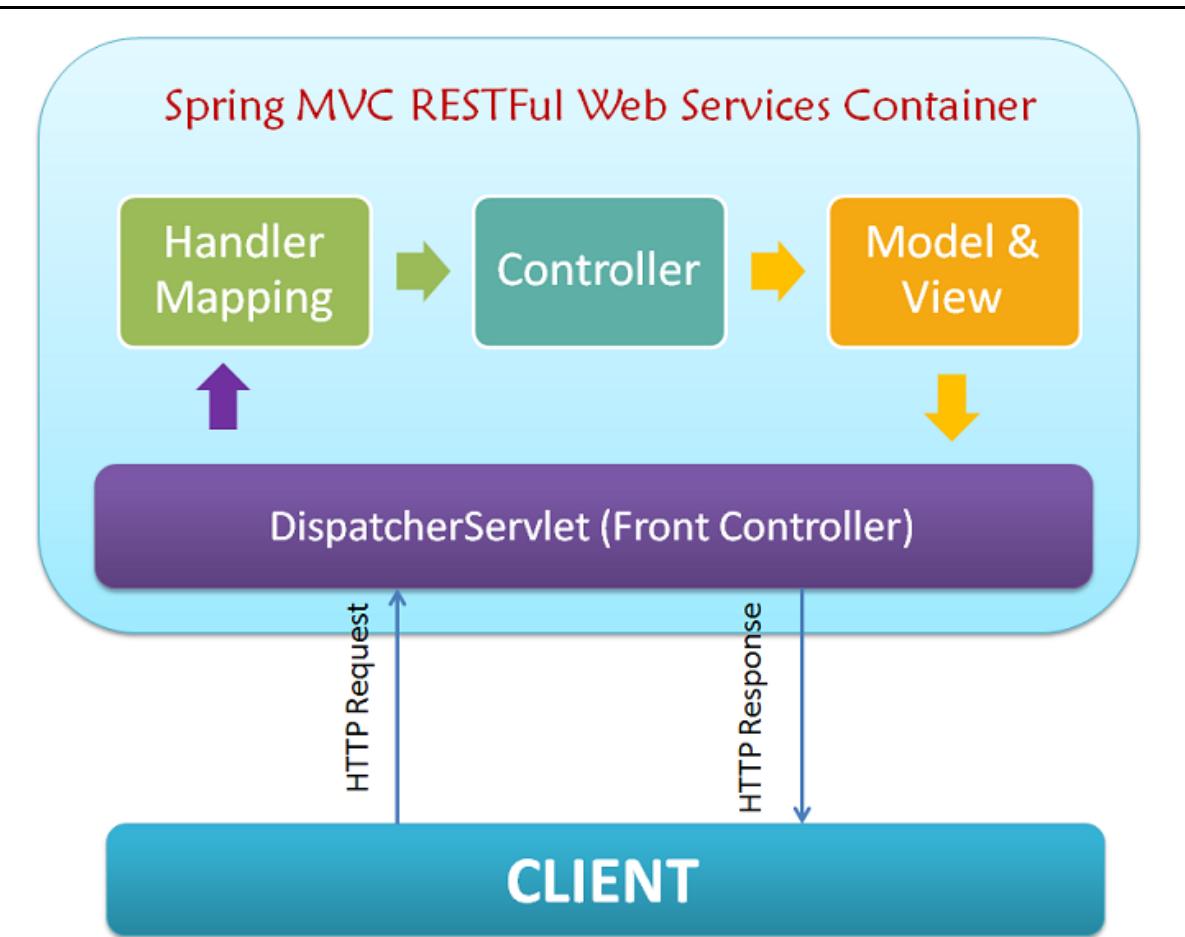


Figure 156:Dispatcher Servlet

Dispatcher servlet needs a configuration file to understand which request needs which controller. Spring mvc provides annotations and all controllers should be annotated with the help of controllers. Every controller will have some mapping.

When the client does the request, the request will go through the web.xml file to the dispatcher servlet. Dispatcher controllers have some configuration and know which request needs which controller and by helping annotation. After doing some logic their controller will send the name of viewer to the dispatcher servlet again as a return. And it will call the particular view. And response will send it to the client.

According to this, everything done by the dispatcher servlet.

Because of this technology even if we change the view technology (thymeleaf, velocity, freemarker) to another technology it will not affect controllers. Just want to change the configuration file.

In spring even if we create a small application, we can expand it very easily.

If we try to develop a **Spring** project, we still have to put in a lot of effort. But **Spring boot** is introduced to **save time to bootstrap the spring projects**. Spring boot is a way to create an application with zero or minimal configurations. It provides a lot of defaults and configuration to create Spring applications faster.

Spring framework

Spring framework is the most widely used Java framework for building applications. Spring framework has a lot of projects to develop java applications.

It has good features like dependency injection and modules such as

- Spring MVC
- Spring Security
- Spring AOP
- Spring ORM
- Spring data

Spring vs Spring boot

We will see how Spring boot can reduce your efforts to bootstrapping any spring application.

Hibernate

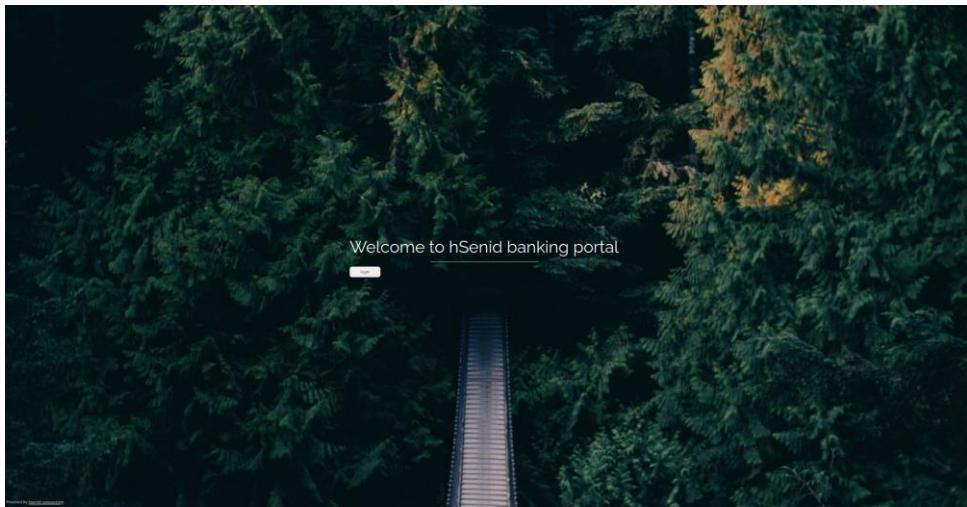


Figure 157: Hibernate bankapp

Login Page

Please provide your credential to use this website

User Id:

Password:

New User? [Register Here](#)

Tecno

Figure 158: Hibernate login

hSenidBank Registration Form

First Name	<input type="text"/> (max 30 characters a-z and A-Z)
Middle Name	<input type="text"/> (max 30 characters a-z and A-Z)
Last Name	<input type="text"/> (max 30 characters a-z and A-Z)
Email	<input type="text"/>
User ID	<input type="text"/>
Password	<input type="password"/>

Figure 159:Hibernate registration

After registering with the system, System will prompt a message (Hibernate Understanding, n.d.).

Thanks for Registering with us :

To login with new Userid and Password[Click here](#)

Figure 160:Thanks for registering

Using your user ID and password, you can login.

If your login ID or password was wrong. It will prompt a message called your login was unsuccessful.

Your Login Was Unsuccessful - Please Try Again

To login again [click here](#).

Figure 161:Login was unsuccessful

If your login information is true, you will be redirected to the home interface.

You can see your sensitive information there. and you will be able to withdraw or deposit money

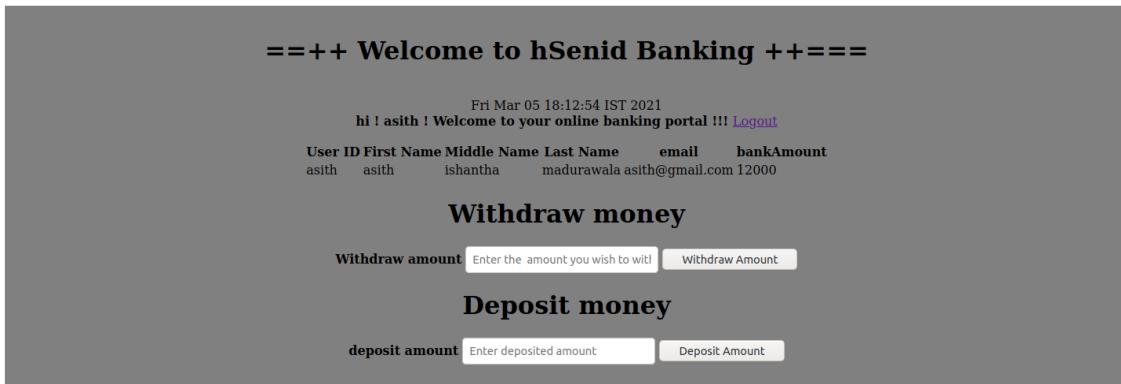


Figure 162:homepage

your current balance is : 13200

If you want more services ... [click here](#). To login again ... [click here](#). To logout ... [click here](#).

Figure 163:After withdraw money UI interface

Spring Boot Training

Rest API's controllers in bank application. It contains 5 controllers.

`@GetMapping("/users")`

Used to get all users that saved in database.

@PostMapping("/users")
Post request to save new user.

@GetMapping("/users/{id}")
Used to get selected user using userID

@PutMapping("/users/{id}")
Used to update selected user using UserID

```
@RestController
@RequestMapping("/api")
public class UserController {

    @Autowired
    RestTemplateClass restTemplate;

    @Autowired
    UserRepository userRepository;

    @GetMapping("/users")
    public List<User> getAllUsers() { return userRepository.findAll(); }

    @PostMapping("/users")
    public User createUser(@Valid @RequestBody User user) { return userRepository.save(user); }

    @GetMapping("/users/{id}")
    public User getUserId(@PathVariable(value = "id") Long userId) {
        return userRepository.findById(userId)
            .orElseThrow(() -> new ResourceNotFoundException("User", "id", userId));
    }
}
```

Figure 164:REST controller spring boot

```
@PutMapping("/users/{id}")
public User updateUser(@PathVariable(value = "id") Long userId,
                      @Valid @RequestBody User userDetails) {

    User user = userRepository.findById(userId)
        .orElseThrow(() -> new ResourceNotFoundException("User", "id", userId));

    user.setFirstName(userDetails.getFirstName());
    user.setMiddleName(userDetails.getMiddleName());
    user.setLastName(userDetails.getLastName());
    user.setEmail(userDetails.getEmail());
    user.setUserId(userDetails.getUserId());
    user.setPassword(userDetails.getPassword());
    user.setBankAmount(userDetails.getBankAmount());

    User updatedUser = userRepository.save(user);
    return updatedUser;
}

@DeleteMapping("/users/{id}")
public ResponseEntity<?> deleteUser(@PathVariable(value = "id") Long userId) {
    User user = userRepository.findById(userId)
        .orElseThrow(() -> new ResourceNotFoundException("User", "id", userId));

    userRepository.delete(user);

    return ResponseEntity.ok().build();
}
```

Figure 165:REST controller spring boot2

hire-purchase module is an extension to the Bank application developed using Spring boot.

It is an additional module developed using TestDrivenDevelopment approach

Mortgage calculator Form is an application that used spring boot technology to develop the application. Not only springboot technology, I used TestDrivenDevelopment to build this application.

Mortgage Calculator Form

- The 'paymentType' of '0' means payment is at the beginning of the period and a '1' means that the payment is at the end of the period.
- The 'futureValue' is your desired balance at the end of the duration.
- Monthly payment will be calculated on the next page.

The screenshot shows a form titled "Mortgage Calculator Form". It contains the following fields:

- start Date (MM/dd/yyyy): 03/05/2021
- interest Rate (%): 4.25
- initial Balance (\$): 300000.00
- duration In Months: 360
- future Value (\$): 0.00
- payment Type (0 or 1): 0

At the bottom are two buttons: "Calculate" and "Reset Fields".

Figure 166:Hire purchase module

When enter the Interest rate, Initial balance and the DurationIn Months that expect to pay, This system generated installment values with respect to Interest rate.

Parameter	Value
startDate	03/05/2021
interestRate	6.55%
initialBalance	\$300,000.00
durationInMonths	12
futureValue	\$0.00
paymentType	0
monthlyPayment	(\$25,895.83)

[Reset](#)

Amortization Schedule

paymentNumber	paymentDate	balance	principalPaid	interestPaid	accumulatedInterest
1	04/05/2021	\$275,741.67	(\$24,258.33)	(\$1,637.50)	(\$1,637.50)
2	05/05/2021	\$251,350.93	(\$24,390.74)	(\$1,505.09)	(\$3,142.59)
3	06/05/2021	\$226,827.06	(\$24,523.87)	(\$1,371.96)	(\$4,514.55)
4	07/05/2021	\$202,169.32	(\$24,657.73)	(\$1,238.10)	(\$5,752.64)
5	08/05/2021	\$177,377.00	(\$24,792.32)	(\$1,103.51)	(\$6,856.15)
6	09/05/2021	\$152,449.35	(\$24,927.65)	(\$968.18)	(\$7,824.34)
7	10/05/2021	\$127,385.64	(\$25,063.71)	(\$832.12)	(\$8,656.45)
8	11/05/2021	\$102,185.13	(\$25,200.52)	(\$695.31)	(\$9,351.77)
9	12/05/2021	\$76,847.06	(\$25,338.07)	(\$557.76)	(\$9,909.53)
10	01/05/2022	\$51,370.68	(\$25,476.37)	(\$419.46)	(\$10,328.99)
11	02/05/2022	\$25,755.25	(\$25,615.43)	(\$280.40)	(\$10,609.38)
12	03/05/2022	\$0.00	(\$25,755.25)	(\$140.58)	(\$10,749.96)

[Reset](#)

Figure 167:Hire purchase module payment schedule

Frontend development: Angular

In this application, I have used spring boot rest API controller as the backend for this application which consists bank logics.

Angular work as the front end for the bank application logic.

Spring boot backend rest API logic works as the backend for the bank application.

```
import java.util.List;

@SpringBootApplication
@RestController
@CrossOrigin(origins = "*")
public class RegistrationServiceApplication {

    @Autowired
    private UserRepository repository;

    @PostMapping("/register")
    public String register(@RequestBody User user) {
        repository.save(user);
        return "Hi " + user.getName() + " your hSenid portal Registration process successfully completed";
    }

    @GetMapping("/getAllUsers")
    public List<User> findAllUsers() { return repository.findAll(); }

    @GetMapping("/findUser/{email}")
    public List<User> findUser(@PathVariable String email) { return repository.findByEmail(email); }

    @DeleteMapping("/cancel/{id}")
    public List<User> cancelRegistration(@PathVariable int id) {
        repository.deleteById(id);
        return repository.findAll();
    }

    public static void main(String[] args) { SpringApplication.run(RegistrationServiceApplication.class, args); }
}
```

Figure 168:Spring boot backend rest API logic

```

@CrossOrigin(origins = "http://localhost:4200")
@org.springframework.web.bind.annotation.RestController
public class RestController {

    @Autowired
    private UserService userService;

    @GetMapping("/")
    public String hello(){
        return "this is home page";
    }

    @PostMapping("/register")
    public String Register(@RequestBody User user){
        userService.saveUser(user);
        return "hi"+ user.getFirstName() + "your registration process sucessfully completed.";
    }

    @GetMapping("/finduser/{id}")
    public List<User> findUser(@PathVariable String id){
        return (List<User>) this.userService.findByBankId(id);
    }
}

```

Figure 169:Spring boot backend rest API logic2

```

@GetMapping("/finduser/{id}")
public List<User> findUser(@PathVariable String id){
    return (List<User>) this.userService.findByBankId(id);
}

@GetMapping("/allusers")
public List<User> getUsers(){
    return this.userService.showAllUsers();
}

@GetMapping("/saveuser")
public String saveUser(@RequestParam String firstName,@RequestParam String age,@RequestParam String bankaccountNumber,@RequestParam String bankAmount,@RequestParam String password){
    User user = new User(firstName,age,bankaccountNumber,bankAmount,bankId,password);
    userService.saveUser(user);
    return "user saved";
}

@DeleteMapping("/delete/{id}")
public List<User> deleteUser(@PathVariable int id){
    return this.userService.deleteMyUser(id);
}

```

Figure 170:Spring boot backend rest API logic3

This is the Angular frontend for the bank application.

User List						
id	firstName	age	bankaccountNumber	bankAmount	bankId	password
4	shohi	23	123456	55000	13	1995
5	ameko	21	546576	85000	13	1998
6	john	12	12345	5500	5	2000
7	jonny	43	123456	12345	67	12345
8	joe	43	12345	67000	21	12
9	jhonny	12	1234	21	12	12
11	asithishantha	12	1995044911	34000	13243	123
12	john	12	19952033911	34000	2432	123
13	jack	12	19950232911	34000	12424	123
14	asithishantha	12	1995044911	34000	13243	123
15	john	12	19952033911	34000	2432	123
16	jack	12	19950232911	34000	12424	123
17	asithishantha	12	1995044911	34000	13243	123
18	john	12	19952033911	34000	2432	123
19	jack	12	19950232911	34000	12424	123
20	asithishantha	12	1995044911	34000	13243	123

Figure 171:Angular frontend

2.3 Tools, Technologies and Applications

- IntelliJ IDEA
 - IntelliJ IDEA is a special programming environment or integrated development environment (IDE) largely meant for Java. This environment is used especially for the development of programs. It is developed by a company called JetBrains, which was formerly called IntelliJ. It is available in two editions: the Community Edition which is licensed by Apache 2.0, and a commercial edition known as the Ultimate Edition. Both of them can be used for creating software which can be sold. What makes IntelliJ IDEA so different from its counterparts is its ease of use, flexibility and its solid design.

- Spring boot

- Spring Boot is an open-source micro framework maintained by a company called Pivotal. It provides Java developers with a platform to get started with an auto configurable production-grade Spring application. With it, developers can get started quickly without losing time on preparing and configuring their Spring application.
 - Java
- Java is a write-once, run-anywhere programming language developed by Sun Microsystems. It is similar to C and C++ but a lot easier. You can combine Java with a lot of technologies like Spring, node JS, Android, Hadoop, J2EE, etc.... to build robust, scalable, portable and distributed full-fledged applications. Java also promotes continuous integration and testing using tools like Selenium.
 - Angular 8.3
- Angular is a TypeScript-based open-source web application framework led by the Angular Team at Google and by a community of individuals and corporations.
 - MongoDB
- MongoDB is a document-oriented NoSQL database used for high volume data storage. MongoDB is a database which came into light around the mid-2000s. It falls under the category of a NoSQL database.
- Git Lab
 - Gitlab is a service that provides remote access to Git repositories. In addition to hosting your code, the services provide additional features designed to help manage the software development lifecycle. These additional features include managing the sharing of code between different people, bug tracking, wiki space and other tools for 'social coding'.
 - You are going to use Gitlab to handle all levels of Java and Non-Java projects. So, it will be good if you have some amount of exposure to software development life cycle and working knowledge of developing web-based and non-web-based applications, along with usage of command prompts on Windows or Linux environments.
- Postman Application
 - Postman is currently one of the most popular tools used in API testing. It started in 2012 as a side project by Abhinav Asthana to simplify API workflow in testing and development. API stands for Application Programming Interface which allows software applications to communicate with each other via API calls.
- Maven
 - Maven is an automation and management tool developed by Apache Software Foundation. It was initially released on 13 July 2004. In Yiddish language the meaning of Maven is "accumulator of knowledge".

- It is written in Java Language and used to build and manage projects written in C#, Ruby, Scala, and other languages. It allows the developer to create projects using Project Object Model and plugins.
- It helps to build projects, dependency, and documentation. Its development process is very similar to ANT. However, it is much more advanced than ANT.

- Bash Scripts

- Plain text files in Linux which contain a bunch of commands which are typically in the command line. Instead of running those commands can save those commands in the bash file and can execute that bash file.

- JUnit testing

- In JUnit, testing contains a class with testing methods. It's a testing framework available for Java. Mainly used for unit Testing to execute a specific functionality of code.

- MYSQL

- In JUnit, testing contains a class with testing methods. It's a testing framework available for Java. Mainly used for unit Testing to execute a specific functionality of code.

- Tomcat

- Apache Tomcat (called "Tomcat" for short) is an open-source implementation of the Java Servlet, Java Server Pages, Java Expression Language and WebSocket technologies. (How to Install Tomcat 9 on Ubuntu 18.04, n.d.)Tomcat provides a "pure Java" HTTP web server environment in which Java code can run.
- Tomcat is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation, released under the Apache License 2.0 license.

- JSP

- JSP technology is used to create web applications just like Servlet technology. It can be thought of as an extension to Servlet because it provides more functionality than servlet such as expression language, JSTL, etc.
- A JSP page consists of HTML tags and JSP tags. The JSP pages are easier to maintain than Servlet because we can separate designing and development. It provides some additional features such as Expression Language, Custom Tags, etc.

- Spring MVC

- A Spring MVC is a Java framework which is used to build web applications. It follows the Model-View-Controller design pattern. It implements all the basic features of a core spring framework like Inversion of Control, Dependency Injection.
- A Spring MVC provides an elegant solution to use MVC in spring framework by the help of Dispatcher Servlet. Here, Dispatcher Servlet is a class that receives the incoming request and maps it to the right resource such as controllers, models, and views.

- Hibernate

- Hibernate ORM (or simply Hibernate) is an object-relational mapping tool for the Java programming language. It provides a framework for mapping an object-oriented domain model to a relational database. Hibernate handles object-relational impedance mismatch problems by replacing direct, persistent database accesses with high-level object handling functions (Hibernate Understanding, n.d.).

- TDD Development

- Test-driven development (TDD) is a software development process relying on software requirements being converted to test cases before software is fully developed, and tracking all software development by repeatedly testing the software against all test cases. This is opposed to software being developed first and test cases created later.
- American software engineer Kent Beck, who is credited with having developed or "rediscovered" the technique, stated in 2003 that TDD encourages simple designs and inspires confidence.

- Python

- Python is a Cobra replica automobile company based in Melbourne, Australia. Founded in 1981, this company has no relation to the more widely known Python automobile created in the 1980s by Kelly Motors at Riverside, California and later at Fort Collins, Colorado.

- Beautiful Soap

- Beautiful Soup is a Python package for parsing HTML and XML documents (including having malformed markup, i.e., non-closed tags, so named after tag soup). It creates a parse tree for parsed pages that can be used to extract data from HTML, which is useful for web scraping (Beautiful soup, n.d.).
 - Selenium
- Selenium is a portable framework for testing web applications. Selenium provides a playback tool for authoring functional tests without the need to learn a test scripting language (Selenium IDE). It also provides a test domain-specific language (Selenese) to write tests in a number of popular programming languages, including C#, Groovy, Java, Perl, PHP, Python, Ruby and Scala.
- Pebble Template
- Pebble is a Java templating engine inspired by Twig and similar to the Python Jinja Template Engine syntax. It features templates inheritance and easy-to-read syntax, ships with built-in autoescaping for security, and includes integrated support for internationalization.
- Java FX
- JavaFX is a software platform for creating and delivering desktop applications, as well as rich Internet applications (RIAs) that can run across a wide variety of devices. JavaFX is intended to replace Swing as the standard GUI library for Java SE, but both will be included for the foreseeable future.

2.4 Work and Developments during the internal training period

2.41 Applova Testing

1. Merchant side category testing

First I started testing with the merchant side category field. I created a category called 屋台食べ物。

<input type="checkbox"/> たこ焼き	<input type="checkbox"/> サブカテゴリ	<input checked="" type="checkbox"/> 食べ物	<input checked="" type="checkbox"/> 有効	<input type="checkbox"/> 編集	<input type="checkbox"/> 削除	<input type="checkbox"/> 無効化	<input type="checkbox"/> 操作
<input type="checkbox"/> 屋台食べ物	<input checked="" type="checkbox"/> カテゴリー		<input checked="" type="checkbox"/> 有効	<input type="checkbox"/> 編集	<input type="checkbox"/> 削除	<input type="checkbox"/> 無効化	<input type="checkbox"/> 操作
<input type="checkbox"/> 食べ物	<input checked="" type="checkbox"/> ハンバーガー		<input checked="" type="checkbox"/> 有効	<input type="checkbox"/> 編集	<input type="checkbox"/> 削除	<input type="checkbox"/> 無効化	<input type="checkbox"/> 操作

Figure 172:created a category called 屋台食べ物。

After that i pressed the **編集** to edit **カテゴリー詳細の編集** (category detail editing) and **カテゴリー画像の編集**(category editing image).



Figure 173:pressed the **編集**

When press the **カテゴリー詳細の編集**. Prompt another window that must contains category info. I filled text boxes and check boxes regarding category.



Figure 174:fill information regarding category

When press **カテゴリー画像の編集**, i was able to add image regarding product(商品に関する画像).



Figure 175:商品に関する画像).

When press, **更新(update)**, i was able to add image related to category successfully.



Figure 176:press, 更新(update),

I have tried once to delete the category using **削除 ボタン**. and it worked well. I was able to delete the category.

When press **無効化** feature provides unavailability for the category.
Before do the unavailability there is a prompt message for the confirmation,



Figure 177:press 無効化 feature provides unavailability for the category

When we press the same button again, we must be able to see the availability of categories

We can see the eye icon change when pressing it and change it according to availability.



Figure 178:we press the same button again, we must be able to see the availability of categories



Figure 179:confirmation of the re availability

I was able to do the re-availability of categories.



Figure 180: Confirmation of re availability of item

There was a message also displayed for the confirmation of updates.



Figure 181: confirmation of updates

操作 function provides two facilities.

1. Creating sub categories サブカテゴリーの追加
1. Commodity display order change 商品の表示順の変更

When press サブカテゴリーの追加, i was able to create sub category called お好み焼き。



Figure 182:1. Commodity display order change 商品の表示順の変更

I created a sub category called お好み焼き here and press(add) to add it into sub topic.



Figure 183: created a sub category called お好み焼き here and 追加(add) press to add it into sub topic.

After 更新 press this. We will be able to see it as a subcategory of 屋台食べ物 category.

<input type="checkbox"/> 屋台食べ物	<input type="checkbox"/> サブカテゴリー	<input checked="" type="checkbox"/> お好み焼き	<input checked="" type="checkbox"/> 有効	<input type="checkbox"/> 編集	<input type="checkbox"/> 削除	<input type="checkbox"/> 無効化	▼ 操作
<input type="checkbox"/> 会員登録							

Figure 184: After 更新 press this. We will be able to see it as a subcategory of 屋台食べ物 category.

1. Merchant side item testing

After the category section. I went through the 商品, see each item features.

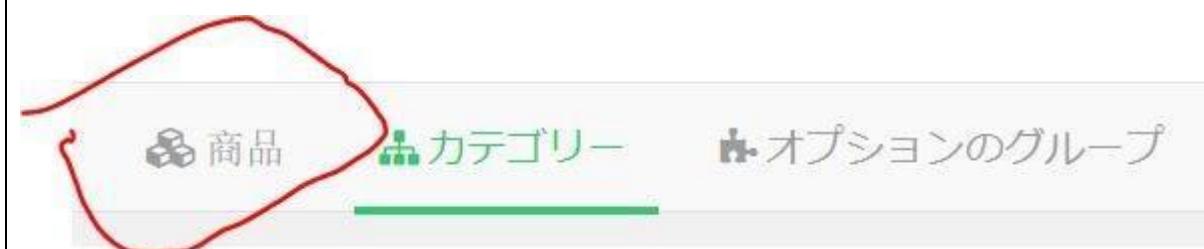


Figure 185:Item section

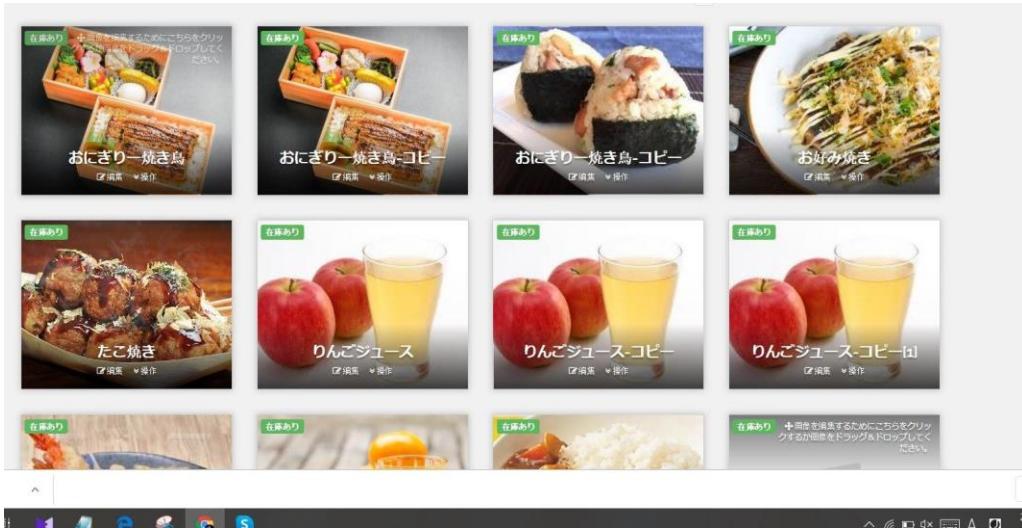


Figure 186:Available items



Figure 187:Takoyaki japanese food

I took one item to see their features. Each item consists of two functions.

1.編集

2.操作

When press 編集 button, I was able to do change the description, prices and tags for each item.



Figure 188:change the description



Figure 189:confirmation of change the description

When press 操作 button,

在庫切れの変更- function gives out of stock change to item.
Then it becomes, out of stock unavailability.



Figure 190:make carbon copy of the item

商品のコピー - function gives functionality, of becomes carbon copy of item.



Figure 191:two items

オプションのグループ管理(option group management)—function gives functionality of adding add-on option of different items like cheese, salt and sugar. We can give these add on option for each item in 商品 according to desire and food type. When buyers buy each product, they have freedom to use these add-on options.

+ オプションのグループ

The screenshot shows a modal dialog titled '+ オプションのグループ'. It contains a table with three columns: 'ドオングループ' (Option Group), 'アドオンオプション' (Add-on Option), and '選択済み' (Selected). There are three rows in the table:

ドオングループ	アドオンオプション	選択済み
のり	のり1 のり2	<input checked="" type="checkbox"/>
チーズ	チーズ1 チーズ2	<input checked="" type="checkbox"/>
塩	塩	<input checked="" type="checkbox"/>

At the bottom right are two buttons: 'キャンセル' (Cancel) and '更新' (Update).

Figure 192: オプションのグループ管理(option group management)

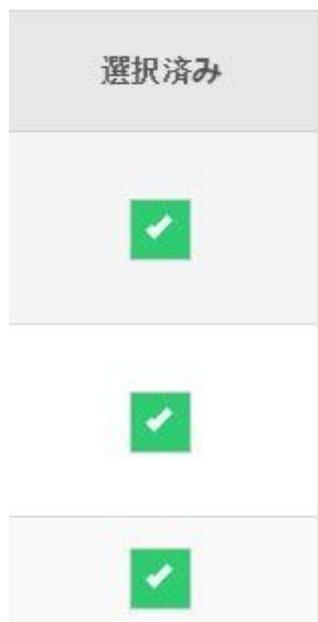


Figure 193: finished selection

ドングループ	アドオンオプション	必要性	ステータス	操作
のり	のり1 のり2	はい	有効	<input type="checkbox"/> 編集 <input type="checkbox"/> 削除
チーズ	チーズ1 チーズ2	はい	有効	<input type="checkbox"/> 編集 <input type="checkbox"/> 削除
塩	塩	はい	有効	<input type="checkbox"/> 編集 <input type="checkbox"/> 削除

Figure 194:Add on option for food



Figure 195:ask to add more images

This function gives the feature of adding more images regarding food items.

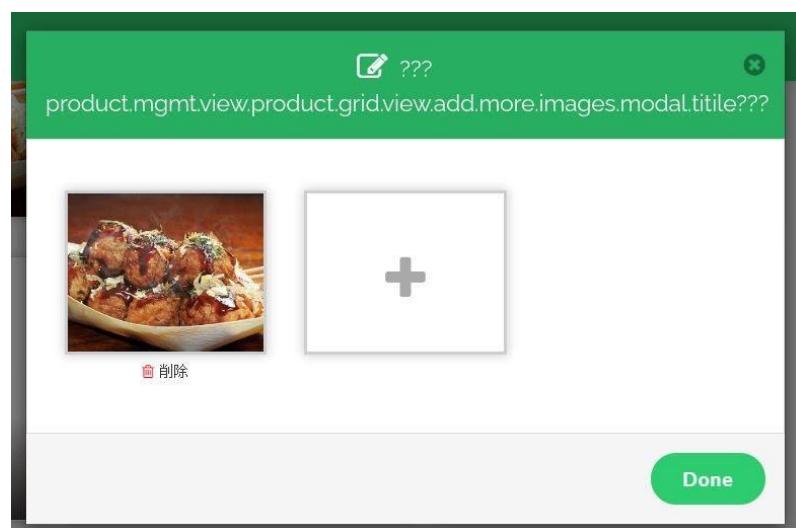


Figure 196:add images location

Delete existing images or add new images can be done using this function related to the selected item.

Adding a new item consists of 3 stages.

1. 基本設定 - foundation establishment

2. 画像追加 - picture addition

3. 詳細設定 - details establishment

基本設定 - foundation establishment

Is a description about each item, information, prices and related tags.

I added a new item called **すき焼き**. I added a description about **sukiyaki** and added price and related tag for this item.

After pressing (**更新**) button, I was able to add new items successfully.

更新を押すと



Figure 197:adding new items location

商品インポート functionality provides feature of import new item instead of creating it.

Using the csv template file, we can do the import of new items.
I am hoping to do this testing in later inspection.

In **商品管理** (item management) section, there is a item search functionality.

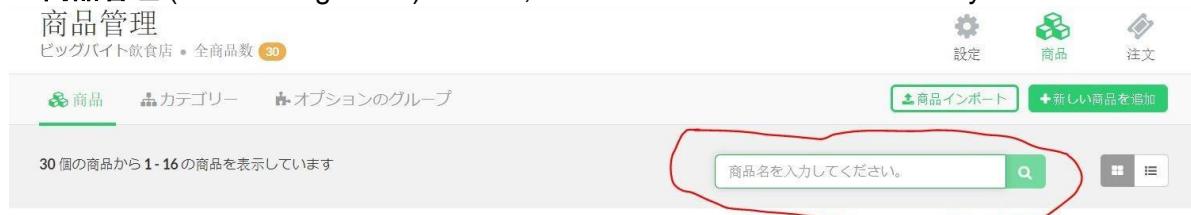


Figure 198: item search functionality.

When カレーライス as カレー、it works well without error.



Figure 199:search curry item

When i tried to search food items, which contain 焼き in food item name, this function was able to display all items the has yaki with their name.

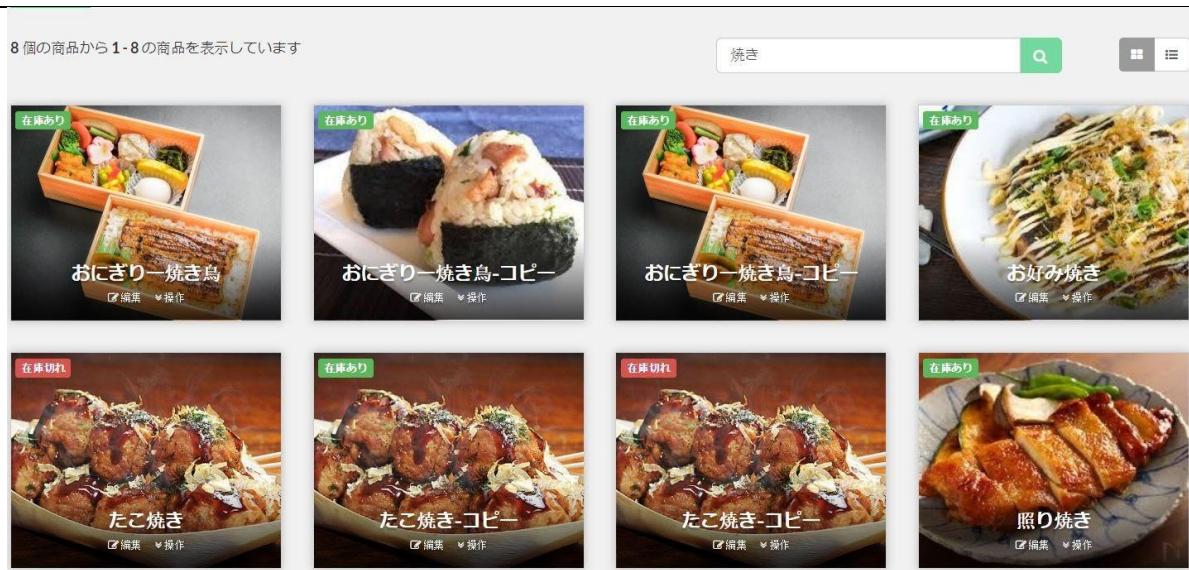


Figure 200: Display all items

ビッグバイト飲食店- customer side

When I login to Applova web application customer portal, the site looks like this.

Figure 201: customer side main interface

When I clicked an item, it opened a new window with description of that item.
There is a feature called **add to cart**(カートに入れる).

As desired we can add any items to the cart without limit.



Figure 202:add items without limit

According to the selected item, there can be an add-on option of adding salt, cheese, sugar etc....

Note: We identified an error in this window. It is a Japanese language mistake. It is not a functional error.



Figure 203: add on salt, sugar for each food item

When I go inside of my cart, there are many features that are provided to the customer.

すべてを削除する

Figure 204: delete all items in the cart

It provides a feature to delete all items in cart.

There is another feature for a customer to change his order quantity in the cart section. According to quantity change, total prices also changed.



Figure 205:cart

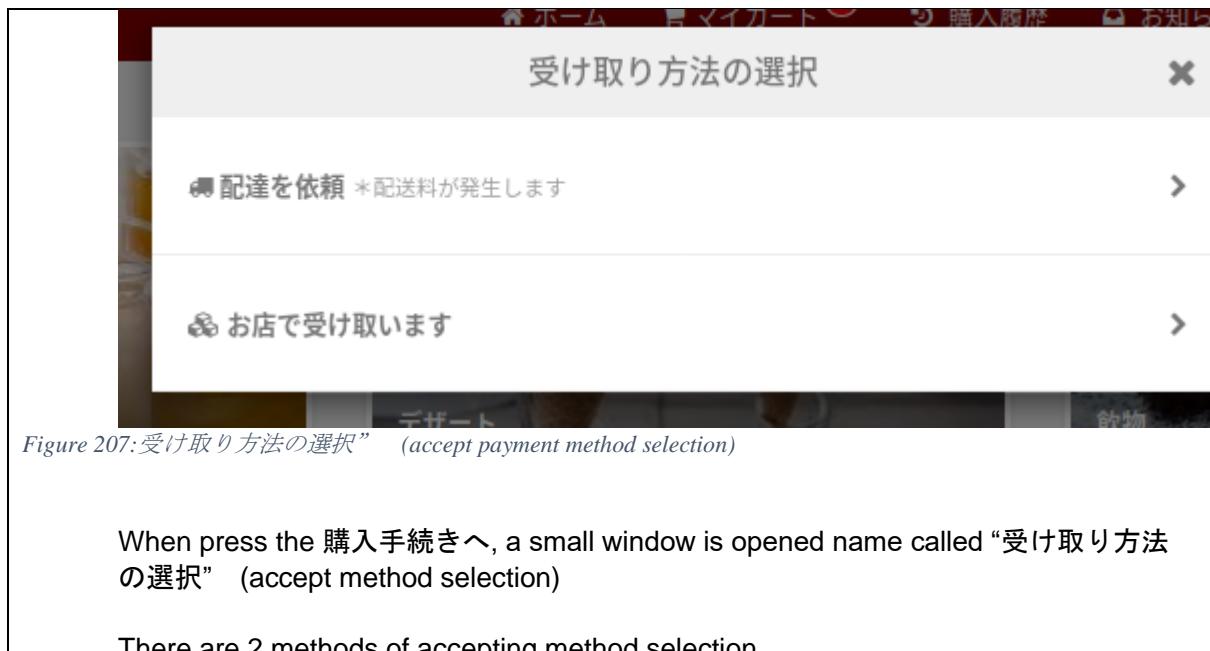
I changed this juice item quantity into 8 from 2 items. According to changes, it changes the total price also.



Figure 206:Cart total price

Cart provides 2 selection options to the customer.

1. 購入手手続きへ
2. ショッピングを続ける



2.42 “HUTCH Danumai Miliyanayai”(Sinhala Tamil Unicode)



Figure 208:HUTCH Danumai Miliyanayai

Sinhala Tamil Unicode is an external collaboration project of Hutch company and hSenid Software International.

BeyondM, a subsidiary of the hSenid Software International is the platform and service partner for “Danumai Miliyanayai”. “BeyondM” supplies software products and services to customers around the world.

myself and my office friend Mr. Dhaneja created 2500 quizzes for “Danumai Miliyanayai 2020” with help of BeyondM people. And translated each question into Singlish, Sinhala and English.

Following are the samples from 2500 quizzes.

United Airlines guwan seva amaerikaa italya hongkong	ಇನಡಿಟಿ ಲೈರ್ಲಿಡೆಸ್‌ಸೆ ಭೂವಿನ ಜೆಲ್‌ಡ್ಯು ನ್ಯೂ ಆರ್ಲಿಂಗ್‌	ಇನಾರ್ಲಿಗ್ ಹೊಂಕೋಂ	Entha n
Ayisanthara duupatha pihta atlantik indiyaaw paesifik	ಅಡಿಯಿನ್‌ಹೆ ದ್ವಾಪನ ಪಿಲೀಂ ಅಂತಿನ್ ಕೂಮಿನ ಒಂಬ್‌ಲ್ಯಾದ್‌ನಿನ್‌	ಉಪ್‌ಲೈ ಪ್ರಾಯ್‌ನಿನ್‌	Iceland
2006 warshaye paapandhu garmaniya indiyaaw arjentinaaw	2006 ವರ್ತತನೆ ಪ್ರಾಪ್‌ನ್‌ಲ್ಯಾಕ್ ಕೂಪಾನ ಶ್ಯಾಲುನಾಕ ಸ್ತರ್‌ನಿನ್‌	ಉನ್‌ಲೈ ಪ್ರಾಪ್‌ನ್‌ನಿನ್‌	2006 Ke
lookayee thunwana vishaal boormiyo sri lanka ayarianthay	ಲೆಲ್‌ನ್‌ಲೈ ಸ್ಯಾವಿನ ವಿಹ್‌ಲೈ ದ್ವಾಪನ ಕೂಲಿಕ ವೆರ್‌ನ್‌ನಿನ್‌	ತ್ರೇ ಲ್ಯಾಕ್ ಅಡಿಲ್‌ಲೈನ್‌	Ulakin r
prasidhha holiwud nalu toi top gan dayi haad prins of thivs	ಪ್ರಸಿದ್‌ದ ಹೈಲೈನ್‌ನ ನಲ್ ತೊಲಿ ಕ್ಯಾಸೆ ಹೆನ್‌ ಎಲ್‌‌ಗನ್‌ ದ್ವಾಪನ ಕೂಲಿಕ ವೆರ್‌ನ್‌ನಿನ್‌	ವೆಡಿ ಹಾವಿ ಪ್ರಿನ್‌ಸ್ ಕೌ ನ್‌ಲೀಂ	Ivatril e
dhumuriyen kelinma lagaawi puththalar kaegagli mulatv	ಡ್ಯುರ್‌ಹಿಯನ್ ಕೆಲಿನ್‌ನ್‌ಲ್ ಲ್ಯಾಕ್ ಖಾಯರ್ ದ್ವಾಪನ ಕೂಲಿಕ ವೆರ್‌ನ್‌ನಿನ್‌	ಹೈಲೈಲ್ ಶ್ಲಾಹ್‌ನ್‌	Ilankai
puerto riko dhuupatha pihi kaeribyan rathu mu kalu muhud	ಪ್ರುಯ್‌ರ್ಲೆ ರ್ಲೋಕ್‌ಎ ದ್ವಾಪನ ತಿನ್‌ಲೀ ಅಂತಿನ್ ಕೂಲಿಕ ವೆರ್‌ನ್‌ನಿನ್‌	ರ್ವ್‌ ತ್ರೀ ಕಾಲ್‌ ಅಷ್‌	Puerto
alitalia guwan samaagama italya malaesiya garmaniya	ಎಲಿಲ್‌ಯಾ ಕೂಲಿಗ್ ದ್ವಾಪನ ಸ್ಯಾಲಾಶಿ ಕೂಮಿನ ರೆಟ್‌ ಆ ಇನ್‌ಲೈ	ಮ್ಲೈಟ್‌ ಸ್ತರ್‌ನಿನ್‌	Entha n
sri lanka nidhahas pakShaj S.W.R.D E Sirimavo W Dahanay nindheneek ಅನ್‌ಹಿತ ಶ್ರೇ ಲ್ಯಾಕ್ ಕಿಲ್‌ಲೈವ್‌ನ್ ಡ್ಯು ಕ್ಲೆಸ್‌ವಿನ್‌ಲ್ಲೀ. ಆ ಸಿರ್‌ಮಾಗ್ ವಿನ್‌ಲೈವ್‌ ದ್ವಾಪನ	ಒನ್‌ಪ್ಲಾಲ್ ಲ್ಯಾಕ್‌ನಿನ್‌ವ್‌	Ilankai s	
Asiana kreeda ulelak pawa indiyawa nepaalay malaesiyaw ಅಕ್ಯಿಯಾನ್ ನ್ಯಿಲ್‌ ದ್ವಾಪನ ಕೂಲಿಗ್ ಅಂತಿನ್ ಕೂಲಿಗ್ ಅಂತಿನ್‌ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	ಒನ್‌ಪ್ಲಾಲ್ ಲ್ಯಾಕ್‌ನಿನ್‌ವ್‌	Ivatril as	
dimokratik pakshaya niyooj bil klintor jor bush ronal reeg ವೆಲ್‌ನ್‌ಲ್ಯಾಕ್‌ನಿನ್‌ ಅಂತಿನ್‌ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	ದ್ವಾಪನ ಕೂಲಿಗ್ ಅಂತಿನ್ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	Intha Av	
looka kusalana papandu th brasiliya singappuri singappuri ಅಕ್ಯಿಯಾನ್ ನ್ಯಿಲ್‌ ದ್ವಾಪನ ಅಂತಿನ್ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	ದ್ವಾಪನ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	Ivatril ka	
Asiana kreeda ulelak pawa iraanaya banglaac singappuri ಅಕ್ಯಿಯಾನ್ ನ್ಯಿಲ್‌ ದ್ವಾಪನ ಅಂತಿನ್ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	ದ್ವಾಪನ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	Ivatril as	
looka kusalana papandu th meksikoo chinaya jaemeikawa ಲೆಲ್‌ನ್ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	ದ್ವಾಪನ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	Ivatril ka	
sayipras dhupatha pihitaa maDyaDh Cheena karibiyan singappuri ಅಕ್ಯಿಯಾನ್ ನ್ಯಿಲ್‌ ದ್ವಾಪನ ಅಂತಿನ್ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	ದ್ವಾಪನ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	Cyprus	
sooviyat dheeeshayata ayatlatiyawa chinaya croeshiyaw ಡೇವೀಲ್‌ ಅಂತಿನ್ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	ದ್ವಾಪನ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	Ivatril m	
1994 - 2001 athara lankaa ranil wika je arya anura baNd 1994 - 2001 ಅಂತಿನ್ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	ದ್ವಾಪನ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	1994-200	
1947 parlimenthuwee vipak aachaaryaa sirimaav E amirthalir 1947 ಲ್ಯಾಕ್ ಅರ್‌ಲೈ ಪ್ರಾಯ್‌ನ್‌ಲೈನ್‌ಎ ವಿಹಿನ್‌ ಆ ಅರ್‌ಲೈ ಲ್ಯಾಕ್ ಅರ್‌ಲೈ ಲ್ಯಾಕ್ ಅರ್‌ಲೈ ಲ್ಯಾಕ್ ಅರ್‌ಲೈ	ದ್ವಾಪನ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	Ilankai	
pancha maha wil etharin ek supiiriyar winipeg viktoriyya 1947 ಲ್ಯಾಕ್ ಅರ್‌ಲೈ ಲ್ಯಾಕ್ ಅರ್‌ಲೈ ಲ್ಯಾಕ್ ಅರ್‌ಲೈ ಲ್ಯಾಕ್ ಅರ್‌ಲೈ ಲ್ಯಾಕ್ ಅರ್‌ಲೈ	ದ್ವಾಪನ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	Intha ee	
pancha maha wil iihita athl Uthru Arr Asiyawa Australiyaw 1947 ಲ್ಯಾಕ್ ಅರ್‌ಲೈ ಲ್ಯಾಕ್ ಅರ್‌ಲೈ ಲ್ಯಾಕ್ ಅರ್‌ಲೈ ಲ್ಯಾಕ್ ಅರ್‌ಲೈ ಲ್ಯಾಕ್ ಅರ್‌ಲೈ ಲ್ಯಾಕ್ ಅರ್‌ಲೈ	ದ್ವಾಪನ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	Periya e	
niwfouland dhuupatha aya caenadaa jaapaani pranshaya 1947 ಲ್ಯಾಕ್ ಅರ್‌ಲೈ	ದ್ವಾಪನ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	Newfou	
prasidhha holiwud nalu toi vaenilaat aytanik shrek 1947 ಲ್ಯಾಕ್ ಅರ್‌ಲೈ	ದ್ವಾಪನ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	Ivatril e	
pancha maha wil etharin ek michigann malawi aemasan 1947 ಲ್ಯಾಕ್ ಅರ್‌ಲೈ	ದ್ವಾಪನ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	Intha ee	
1952 - 1956 athara lankawesi dab aai aachaaan diem iavar 1952 - 1956 ಅಂತಿನ್ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	ದ್ವಾಪನ ಕೂಲಿಗ್ ಅಂತಿನ್ ತಿನ್‌ಲೈವ್‌ನ್ ಡ್ಯು	1952-15	

Figure 209:Danumai Miliyanayai excel sheet

With this

link, can access all questions created by us.

<https://docs.google.com/spreadsheets/d/1Gd23cCSUgeny-NDZmf8INq6IIwB2mOTuulJvwHZWr0Q/edit#gid=1388307343>

2.43 BeyondM messaging platform testing

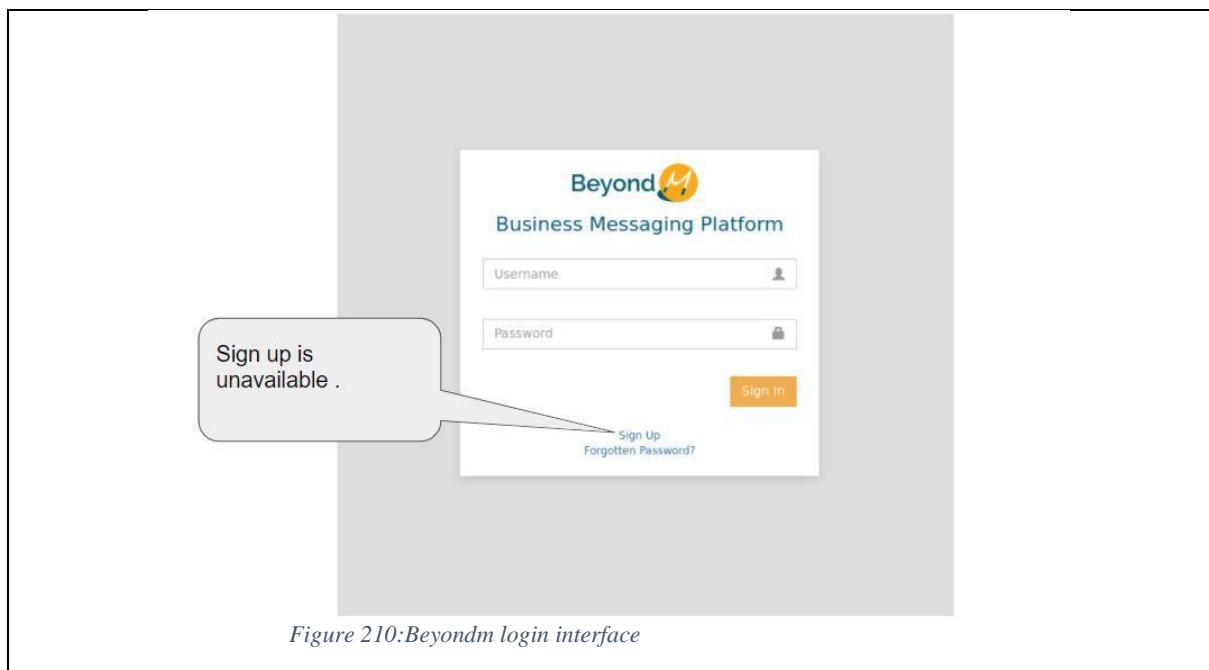


Figure 210:Beyondm login interface

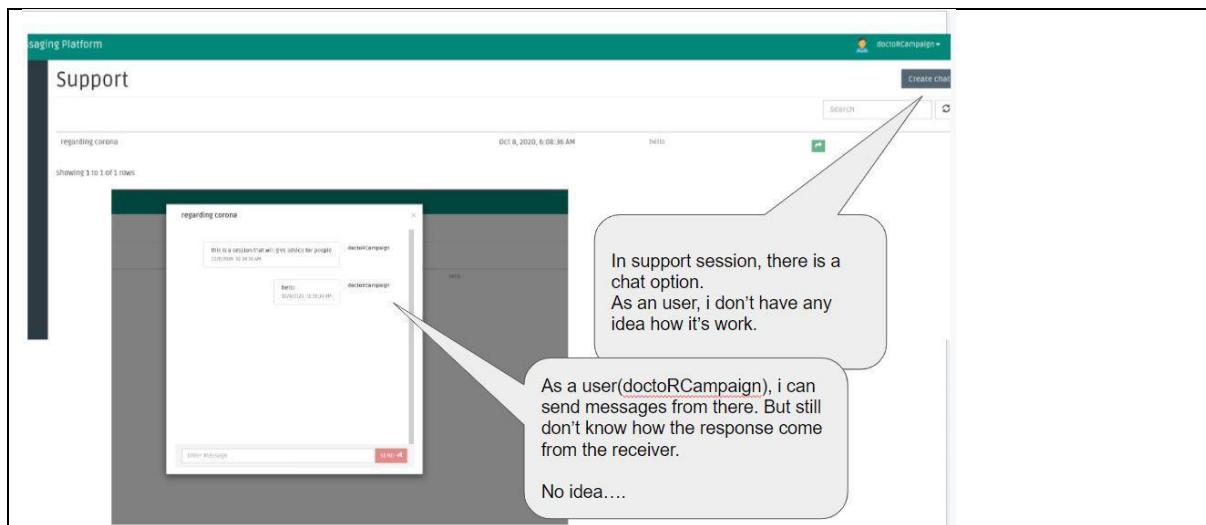


Figure 211:BeyondM support interface

The screenshot shows the 'Subscribers' list from the Business Messaging Platform. It displays two entries: 'MSISON' and '0776532435'. Both entries are marked with a checkbox and the status 'dialing'. A callout box contains the following text:

Hard to identify the meaning.
(user will get lost). User may be
don't know that he/she has to
enter the mobile number of
subscribers there.

Figure 212:Add mobile number of subscribers here

The screenshot shows two views of a messaging platform. On the left, the 'Business Messaging Platform' displays a 'Subscriber Lists' page with one subscriber named 'HypatientsAsiri' (3 subscribers). On the right, the 'SS Messaging Platform' displays a 'Campaign Report' for a campaign named 'Starter'. This report shows 0 subscribers across four scheduled messages. A callout box highlights this inconsistency.

Name	Schedule Time	User	Status	Subscriber List	Subscribers	Success Count	Fail Count	Delivered Count
test	OCT 7, 2020, 4:23:00 PM	merchant_admin	PENDING	077786590,	0	0	0	0
ui	OCT 7, 2020, 8:48:00 PM	merchant_admin	PENDING	abc,	0	0	0	0
doctorcampaign	OCT 8, 2020, 11:58:50 AM	doctorcampaign	PENDING	Hypatientsasiri,	0	0	0	0
Starter	OCT 8, 2020, 4:07:00 PM	merchant_admin	DRAFT	abc,	0	0	0	0

Showing 1 to 4 of 4 rows

Even there are 3 subscribers in user side subscriber list, in Campaign Report it shows that there are 0 subscribers.

Figure 213:BeyondM campaign report

The screenshot shows the 'Business Messaging Platform' displaying a 'Credit Usage Report' for the period from Oct 7, 2020, to Oct 8, 2020. It lists three entries, all labeled 'Plan: Starter'. A callout box asks 'Rs, \$?' about the values. Another callout box notes that 'In admin side, There are DR and CR. Hard to identify of that meaning except admin'.

Date	Description	CR	DR
OCT 7, 2020, 10:36:37 AM	Plan: Starter	500	0
OCT 7, 2020, 2:22:06 PM	Plan: Starter	500	0
OCT 7, 2020, 3:03:06 PM	Plan: Starter	500	0

Showing 1 to 3 of 3 rows

Total of CR : 1500

Total of DR : 0

Rs, \$?

In admin side, There are DR and CR. Hard to identify of that meaning except admin

Figure 214:Credit Usage Report

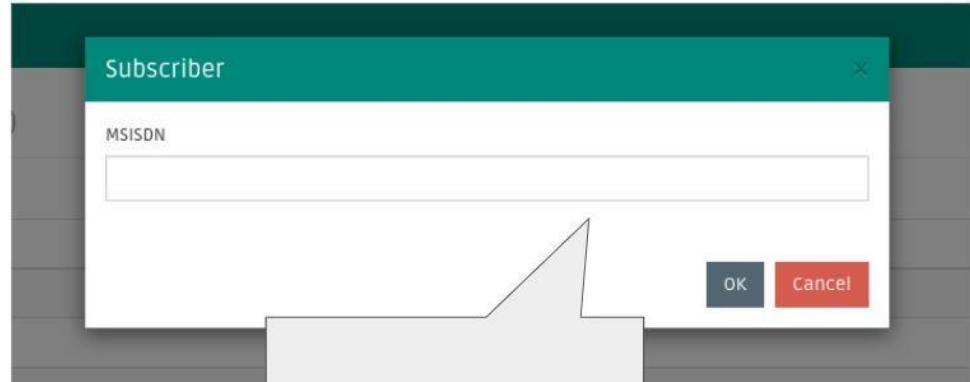
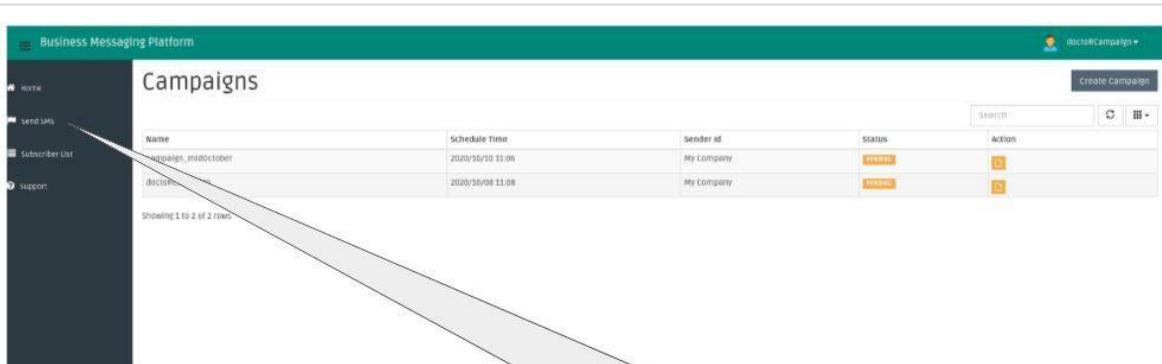


Figure 215:Subscriber information



When press the send sms, it is displayed the campaigns list. As a function name "List campaigns " is appropriate than the "send sms".

Figure 216:send sms interface

The screenshot displays two views of a business messaging platform. The top view is the 'Home' screen, which features a sidebar with links: Home, Send SMS, subscriber list, and Support. The main area shows a summary card with a flag icon, the number '6', and the text 'Scheduled Campaigns'. A callout bubble points from this card to a note: 'There are 6 schedule campaigns are displaying even i had only 2 campaigns in my user account'. The bottom view is the 'Campaigns' screen, showing a table with two rows of campaign details:

Name	Schedule Time	Sender ID	Status	Action
campaign_m10000000000000000000000000000000	2020/01/20 11:00	My Company	Pending	
doctorCampaign	2020/01/09 11:00	My Company	Pending	

Showing 1 to 2 of 2 rows.

Figure 217:schedule the campaigns

2.44 Web Scraping Tabelogu Japan and UberEats Japan



Figure 218:Tabelogu website

hSenid is developing a website called “Applova”, which is very similar to UBER Eats Japan. Restaurants can register with this website and promote and attract customers through this website and customers can order food items from the website and will deliver to the doorstep of customers.

Because of developing a website, which is very similar to Uber eats, Hsenid is looking for the same restaurants’ dataset as UberEats and other similar websites does.



Figure 219:Uber Eats Japan homepage

My task was to obtain all restaurants' information that registered with UberEats Japan and Tabelogu Japan. So, my tech lead advised me to use technology called web scraping to obtain this information from the sites. It was my first experience with web scraping and I had to get a basic idea about what Web Scraping is.

Web scraping is fetching data from a website or webpage. There are different libraries used for implementing code to scraping. Most people use python to write scraping scripts. Some people use Visual basic also.

I decided to use python to write scripts. I went through many libraries and got a basic idea. Each technology has its own features and according to website layout and source code, we can use different technologies.

According to what I understood, I decided to use Python with Beautiful Soup to obtain data from Tabelogu Japan website and Python with selenium to obtain information from UberEats Japan.

Beautiful soup library does get each pages' source code and filter information we are looking for.

Following is the python script that designed to scrape data from Tokyo Area restaurants.

Please use following URL, if want to access the GITHUB source code.

<https://github.com/asithishantha/Tabelogu-Japan-Webscraping-PYTHON-beautifulsoup-git>

```
import requests
from bs4 import BeautifulSoup
import pandas as pd
import csv

baseurl = 'https://tabelog.com/'
url = ' '

headers = {
    'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) '
                  'Chrome/74.0.3729.169 Safari/537.36 '
}
productlinks = []
for x in range(1, 3):
    r = requests.get(f'https://tabelog.com/tokyo/rstLst/cond58-00-00/{x}/?LstSmoking=0&svd=20201013&svt=1900&svps=2')

    # get the first page's URL (page number 1) and identify the location of
    # page number in URL and change the page
```

```

# number as {x} . Because we are going to get all information from all p
ages using for loop.

soup = BeautifulSoup(r.content, 'lxml')

# use pip install lxml

productlist = soup.find_all('h4', class_='list-rst__rst-name')
    # identify the common element cass name. Normally in website, every p
roduct is a object of one class/
        # identify that class name with its tag name.

for item in productlist:
    for link in item.find_all('a', href=True):
        print(link['href'])
        productlinks.append(url + link['href'])

resturantlist = []
for link in productlinks:
    r = requests.get(link, headers=headers)

soup = BeautifulSoup(r.content, 'lxml')

# try:

name = soup.find('h2', class_='display-name').text.strip()
rating = soup.find('b', class_='c-rating_val rdheader-rating_score-
val').text.strip()

number = soup.find('p', class_='rstinfo-table__tel-num-
wrap').text.strip()

address = soup.find('p', class_='rstinfo-table__address').text.strip()

resturant = {

    'name': name,
    'address': address,
    'number': number,
    'ratings': rating
}

```

```

def write_csv(restaurant, url):
    with open('restaurant.csv', 'a' , encoding='utf8') as csvfile:
        writer = csv.writer(csvfile)

        row = [restaurant['name'], restaurant['address'], restaurant['number'], restaurant['ratings'], url]

        writer.writerow(row)

    write_csv(restaurant, link)
    # print(restaurant)

```

Following is the python BeautifulSoup script guidance created by me.

It helps to anyone to run this python script in their own environment.

Webscraping-PYTHON-beautifulsoup-tablelog 食べログ web site

For this project, Python 3.8 will be used.

For Windows installations, when installing Python make sure to check “PATH installation”. PATH installation adds executables to the default Windows Command Prompt executable search. Windows will then recognize commands like “pip” or “python” without requiring users to point it to the directory of the executable (e.g. C:/tools/python/.../python.exe). If you have already installed Python but did not mark the checkbox, just rerun the installation and select modify. On the second screen select “Add to environment variables”.

Getting to the libraries

- * soup is widely used to parse the HTML files

- * Pandas is used to create structured data

To install these libraries, start the terminal of your OS. Type in:

```
pip install BeautifulSoup4 pandas requests lxml
```

WebDrivers and browsers_

Every web scraper uses a browser as it needs to connect to the destination URL. For testing purposes I highly recommend using a regular browser.

To get started, use your preferred search engine to find the “webdriver for Chrome” (or Firefox).

We should begin by defining our browser. Depending on the webdriver we picked back in “WebDriver and browsers” we should type in:

```
headers = {
    'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)'
                  'Chrome/74.0.3729.169 Safari/537.36'
}
```

Finding a cozy place for our Python web scraper

If you already have Visual Studio Code installed, picking this IDE would be the simplest option. Otherwise, I’d highly recommend PyCharm

Importing and using libraries

```
import requests
from bs4 import BeautifulSoup
import pandas as pd
import csv
```

Picking a URL

We are expecting to scrape data from many pages in targeting website. so we have to identify the page number location, which automatically increments in the URL and use {x} for that location. According to the below code, this script will loop 50 times to get information from 50 pages.

```
for x in range(1, 50):
    r = requests.get(f'https://tabelog.com/chiba/rstLst/cond58-00-00/{x}/?LstSmoking=0&svd=20201014&svt=1900&svps=2')

    soup = BeautifulSoup(r.content, 'lxml')
```

Defining objects and building lists

Python allows coders to design objects without assigning an exact type. An object can be created by simply typing its title and assigning a value.

```
# Object is "results", brackets make the object an empty list.  
# We will be storing our data here.  
productlinks = []
```

URL, change the page number location in URL with {x} for support looping 50 times

```
productlinks = []  
for x in range(1, 50):  
  
    r = requests.get(f'https://tabelog.com/chiba/rstLst/cond58-00-  
00/{x}/?LstSmoking=0&svd=20201014&svt=1900&svps=2')  
  
pip install lxml  
  
soup = BeautifulSoup(r.content, 'lxml')
```

In Inspect(F12), identify the tag name or class name of element , which link to the inner page to get more details regarding each product or restaurant. save the each link to productlist.

```
productlist = soup.find_all('h4', class_='list-rst__rst-name')
```

Using for loop, get each restaurant one by one and save into productlink object

```
for item in productlist:  
    for link in item.find_all('a', href=True):  
        print(link['href'])  
        productlinks.append(url + link['href'])
```

Get each product link and direct to the inner page using requests library

```
resturantlist = []  
for link in productlinks:  
    r = requests.get(link, headers=headers)
```

get the content of the inner page using beautifulsoup

```

soup = BeautifulSoup(r.content, 'lxml')

filter the name, rating, number and address of each restaurant's and save as
variables.

find each information using their class name and the tag name

    name = soup.find('h2', class_='display-name').text.strip()
    rating = soup.find('b', class_='c-rating_val rdheader-rating_score-
val').text.strip()

    number = soup.find('p', class_='rstinfo-table_tel-num-
wrap').text.strip()

    address = soup.find('p', class_='rstinfo-table_address').text.strip()

Write into csv file

    def write_csv(restaurant, url):
        with open('restaurant_nagoya.csv', 'a', encoding='utf8') as csvfile:
            writer = csv.writer(csvfile)

                row = [restaurant['name'], restaurant['address'], restaurant['numbe
r'], restaurant['ratings'], url]

            writer.writerow(row)

    write_csv(restaurant, link)

```

Selenium is a webdriver, which is used to test software applications through automation, but we can use selenium for web scraping also.

I used two technologies for two websites. Reason was that the source code of the websites was different. If I used selenium script to automate google.com, it looks easy to scrape.

I created a .csv file from UberEats website restaurants' name and wrote it into csv. Using that .csv file. I was able to scrape data from google.com. I actually didn't scrape data from UberEats directly. I just wrote a python script with selenium to scrape data from google.com.

It was the best solution for me that time.

But when scraping data from Tabelogu Japan website, I was able to get all of the necessary information from Tabelogu website directly. So i wrote a python script to scrape data directly from Tabelogu website using BeautifulSoup.

Following is the Python Script, which designed to scrape data from UberEATS.

I have used Python with selenium driver to automate the google.com to scrape data from UberEATS.

```
# this is a python script, which wrote to get restaurants' details from "Trip Advisor" web site
# still programming

import csv
import openpyxl
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
import time
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC

PATH = "C:\python file\chromedriver.exe"

driver = webdriver.Chrome(PATH)

driver.get("https://www.google.com/")

obj = openpyxl.load_workbook("tokyorestu.xlsx")

# print(obj)

sheet_obj = obj.active
# print(sheet_obj)

m_row = sheet_obj.max_row

for i in range(1, m_row):
    cell_obj = sheet_obj.cell(row=i, column=1)
    restaurantname = cell_obj.value
    driver.get(
        "https://www.tripadvisor.jp/Restaurant_Review-g1066443-d12987949-Reviews-Burger_Milkshake_Crane-Chiyoda_Tokyo_Tokyo_Prefecture_Kanto.html")
    search = driver.find_element_by_class_name("_3qLQ-U8m")
    search.send_keys(restaurantname)
    search.send_keys(Keys.RETURN)

    try:
        main = WebDriverWait(driver, 15).until(
```

```

        EC.presence_of_element_located((By.CLASS_NAME, "location-meta-
block"))
    )

    element = main.find_element_by_class_name("review_count")

    # print(element.text)
    linkmain = WebDriverWait(driver, 10).until(
        EC.presence_of_element_located((By.LINK_TEXT, element.text))
    )

    linkmain.click()

    tabs = driver.window_handles
    for tab in tabs:
        driver.switch_to.window(tab)

    asi = WebDriverWait(driver, 10).until(
        EC.presence_of_element_located((By.CLASS_NAME, "page "))
    )
    address = asi.find_element_by_class_name("_2saB_0Se")

    telephonenumber = asi.find_element_by_xpath("//a/span[1]/span[2]")

    print(address.text, telephonenumber.text)

except:
    print(" exception occurred")

```

You can use following link to access the GITHUB repository project.

https://github.com/asithishantha/UberEats_Japan_Python_Selenium_webscraping_.git

Following is the guidance to run the Python script in any environment.

```

<<<<< HEAD
# web-scraping

=====
#Uber-Eats Web Scraping script Using Selenium in Python

For this project,Python 3.8 will be used.

```

For Windows installations, when installing Python make sure to check “PATH installation”. PATH installation adds executables to the default Windows Command Prompt executable search. Windows will then recognize commands like “pi p” or “python” without requiring users to point it to the directory of the executable (e.g. C:/tools/python/.../python.exe). If you have already installed Python but did not mark the checkbox, just rerun the installation and select modify. On the second screen select “Add to environment variables”.

Getting to the Libraries

- * is widely used to parse the HTML files
- * Pandas is used to create structured data
- * Selenium provides browser automation

To install these libraries, start the terminal of your OS. Type in:

```
pip install BeautifulSoup4 pandas selenium
```

WebDrivers and browsers

Every web scraper uses a browser as it needs to connect to the destination URL. For testing purposes i highly recommend using a regular browser.

To get started, use your preferred search engine to find the “webdriver for Chrome” (or Firefox). Take note of your browser’s current version. Download the webdriver that matches your browser’s version.

```
for chrome driver : https://chromedriver.chromium.org/
```

Finding a cozy place for our Python web scraper

If you already have Visual Studio Code installed, picking this IDE would be the simplest option. Otherwise, I’d highly recommend PyCharm

Importing and using libraries

```
import openpyxl
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
import time
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
```

```
import csv
```

We should begin by defining our browser. Depending on the webdriver we picked back in “WebDriver and browsers” we should type in:

```
driver = webdriver.Chrome(executable_path='c:\\path\\to\\windows\\webdriver\\executable.exe')
```

OR

```
driver = webdriver.Firefox(executable_path='/nix/path/to/webdriver/executable')
```

Picking a URL

Before performing our first test run, choose a URL. As this web scraping tutorial is intended to create an elementary application, I highly recommended picking a simple target URL:

```
driver.get("https://www.google.com/")
```

In this program, we are reading restaurants' name from excel sheet.
to read that excel sheet

```
obj = openpyxl.load_workbook("nagoya.xlsx")
```

after that we have to get restaurants' name from excel sheet one by one
below code is written.

```
sheet_obj = obj.active
# print(sheet_obj)

m_row = sheet_obj.max_row

for i in range(1, m_row):
    cell_obj = sheet_obj.cell(row=i, column=1)
    restaurantname = cell_obj.value # take the restaurant name in first
cell
    driver.get("https://www.google.com/")
    search = driver.find_element_by_name("q") # google search box element
name by name, we can use class name,
    # tag name and any other locate element methods.
    search.send_keys(restaurantname) # search the restaurant name
    search.send_keys(Keys.RETURN) # press search
```

Get elements

To get mobile number from the google. get element by x path is used.

We can use any locating method for that.

To know about locating methods.

<https://www.youtube.com/watch?v=mQ7-mPJYJ5A&t=973s>

If you don't know how to write the x path for the target element, there is a google plugin, which generate the x path for any element .

for more details : <https://www.youtube.com/watch?v=VvZEsZ3cGmc>

```
number2 = driver.find_element_by_xpath("//span[@class='LrzXr zdqRlf kno-fv']/a[1]/span[1]")
```

for write the output data to the csv

```
def write_csv(resturant):
    with open('nagoya_res.csv', 'a', encoding='utf8') as csvfile:
        writer = csv.writer(csvfile)

        row = [resturant['name'], resturant['address'], resturant['number'], url]

        writer.writerow(row)

    write_csv(resturant)
    # write to the csv file. please give the csv file name accordingly, you
    # wish to write the scraped data
    print(resturant)

>>>>> Initial commit
```

2.45 JobsDB testing

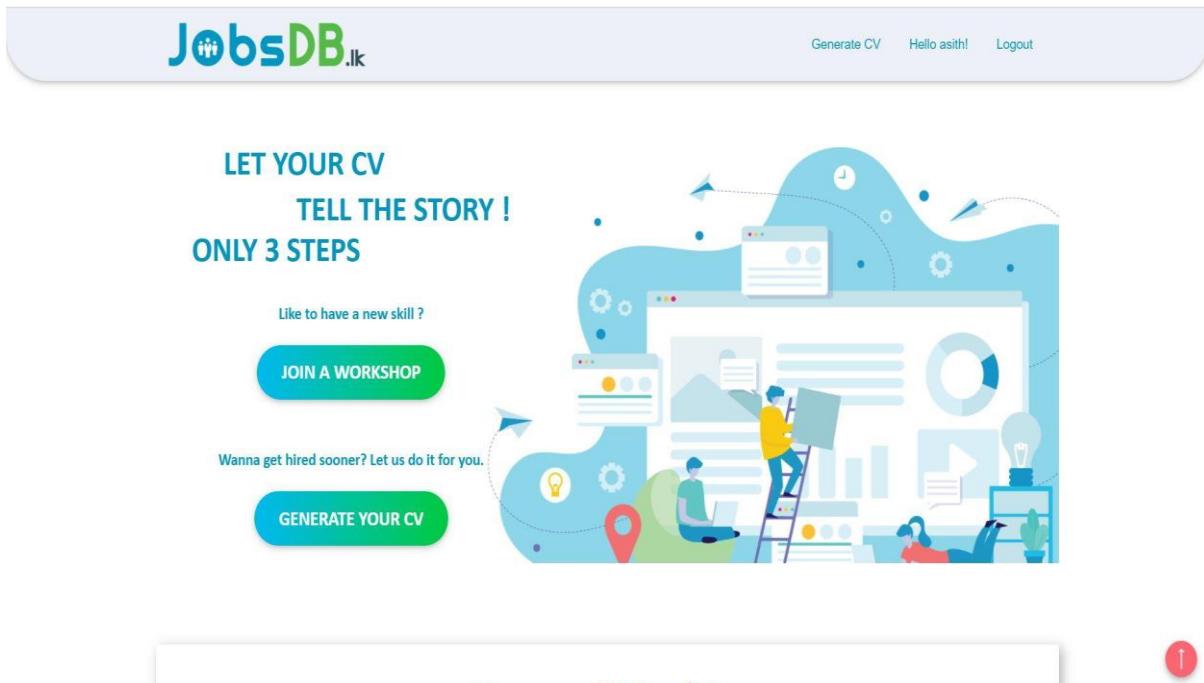


Figure 220:JobsDB homepage

JobsDB is a free cv generation website developed by hSenid Software International as an internal project. Currently it is introduced to many national universities in Sri Lanka.

University of Colombo

University of Ruhuna

University of Peradeniya are few of them.

It includes many cv templates, users can select when creating a cv template.

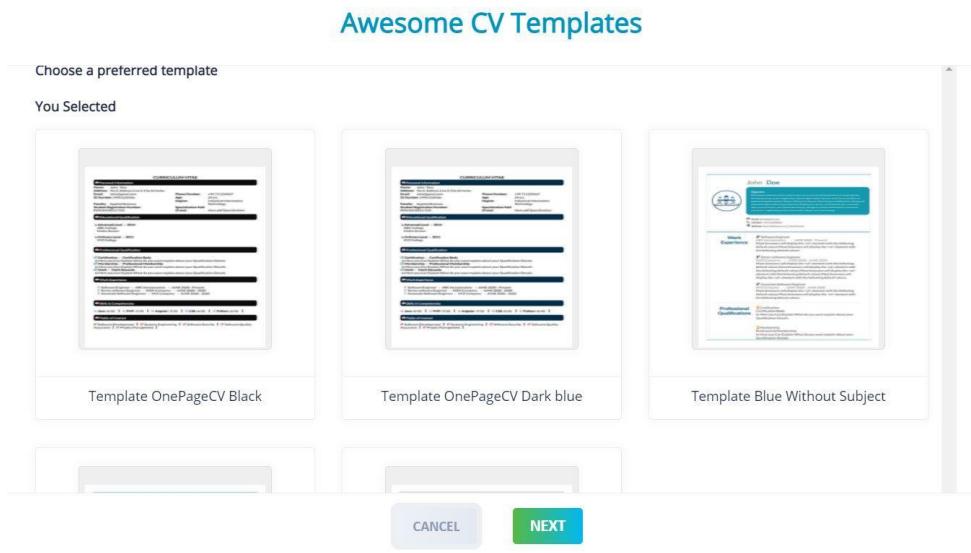


Figure 221:cv templates

My responsibility is as follows in JobsDB.

- create new templates to this website. => used staging website.
- maintain already existing CVs.
- Testing the website

In JobsDB, when talking about jobsdb testing, I tested admin side and user side registration.

I tested the user registration interface with more than 30 test cases as follows.

Can access all test cases through the following link.

https://docs.google.com/spreadsheets/u/5/d/1xDTWr4kkNiegisJbgDP8ZdIETUM6157ZWrIIwB-Hc5A/edit?usp=drive_web&ouid=105036986372973636872

Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R001	Resgister' button click	press Register button	1. go to https://jobsdb.hsenidjapan.hsenid.id/ website 2. press register button	internet connection is needed	load the registration page	registration page is loaded	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R002	Registration page load time	After the press the regier button, register pa	1. press the register button	internet connection is needed	load the registration page within few seconds	registration page is loaded within few seconds	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R003	Input values to fields	enter the true values in the field	1.enter first name2. enter the last name3. enter an email address	you must have an unregistered email address	prompt an message to email validation	prompt an message to email validation	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R004	input true values except firstname	enter the true values in the field except first name	1. ignore first name2. enter the last name3. enter an email address	you must have an unregistered email address	prompt first name is required	prompt first name is required	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R005	input true values except lastname	enter the true values in the field except last name	1.enter first name2. ignore the last name3. enter an email address	you must have an unregistered email address	prompt last name is required	prompt last name is required	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R006	input true values except first name	input true values except first name and last name	1.ignore first name2. ignore the last name3. enter an email address	you must have an unregistered email address	prompt first name is required	prompt first name is required	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R007	input true values except password	input true values except password	1.ignore first name2. ignore the last name3. enter an email address	you must have an unregistered email address	prompt password is required	prompt password is required	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R008	input true values ignoring email, password	input true values ignoring email, password	1.enter first name2. enter the last name3. ignore an email address	you must have an unregistered email address	prompt email, password, re-password	prompt email, password, re-password	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R009	input true values ignoring first name	input true values ignoring first name, last name	1.ignore first name2. ignore the last name3. enter an email address	you must have an unregistered email address	prompt firstname, lastname and past	prompt firstname, lastname and past	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R010	Register' click button with empty field	Register' click button with empty fields	1. ignore all fields2.press register button in register form		prompt all fields fill is required	prompt all fields fill is required	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R011	Clear' button click after filling all field	press the Clear' button in register form	1. enter some values into all field.2. press the clear button	values has to enter all field	clear all fields.	clear all fields.	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R012	Clear' button click after filling first name	Clear' button click after filling first name field	1.enter values into the first name field.2. press clear button.	first name field must be filled	clear first name field.	clear first name field.	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R013	Clear button click after filling last name field	Clear' button click after filling last name field	1.enter values into the last name field.2. press clear button.	lastname name field must be filled	clear last name field.	clear last name field.	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R014	Clear' button click after filling email	Clear' button click after filling email field only	1.enter values into the email field.2. press clear button.	email field must be filled	clear email field.	clear email field.	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R015	Clear' button click after filling re-pass	Clear' button click after filling re-password	1.enter values into the re-password field.2. press clear button.	re-password field must be filled	clear re-password field.	clear re-password field.	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R016	Validating numbers for the first name	Validating numbers for the first name field	1. enter number values into first name field.		prompt "only text required."**	prompt only text required.	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R017	Validating numbers for the last name	Validating numbers for the last name field	1. enter number values into last name field.		prompt "only text required."**	prompt only text required."**	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R018	Validating the email address with es	Validating the email address with es	1.enter text only name or text and numerics combination name with @sign		noting prompt as an error	nothing prompt as an error	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R020	Validating the email address with es	Validating the email address with es	1.enter text only name or text and numerics combination name with @sign		prompt "invalid email format"	prompt "invalid email format"	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R021	Matching the two passwords with va	make sure with valid info.	1. type password in password field.2. type same password (re-password) in re-password field.		nothing prompt as an error.	nothing prompt as an error.	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R022	Matching the two passwords with un	Matching the two passwords with un-match	1. type password in password field.2. type different password (re-password) in re-password field.		prompt an error message called "pa	prompt an error message called "pa	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R023	Cannot register with previously regis	if someone use email for register in this sys	1. fill all fields with necessary info.2. fill email field with already used email address.3. press the		prompt and error called "email alrea	prompt and error called "email alrea	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R024	"Email verification mail sent" messag	if someone register with information and us	1. enter all information.2. press register button.		after pressing the register button,pr	after pressing the register button,pr	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R025	Login without confirming the registr	if user try to login using not confirmed ema	1. login with unregistered email.2. press register button.3.use email and password to login without		prompting message "please confirm	prompting message "please confirm	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R026	verification email receive time	checking verification email receive time.	1. after press registering.2. go to email box.3. wait for the email.		email will receive within 1 minute.	email is received within 1 minute.	pass

user profile	UP001	Select user profile (left side navigation)	selecting the sub title called "User Pr1. Press user profile	you must enter to the dash board	display the user profile interface	display the user profile interface	pass	
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	UP002	update profile first name	updating profile changing first name	1.change fist name2. ignorin	you must enter to the dash board	a prompt message called "succe	a prompt message called "spass	
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	UP003	update profile last name	updating profile changinglast name	1.ignore fist name2. changin	you must enter to the dash board	a prompt message called "succe	a prompt message called "spass	
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	UP004	update mobile number	updating profile changingmobile num	1.ignore fist name2. ignor	you must enter to the dash board	a prompt message called "succe	a prompt message called "spass	
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	UP005	update profile mobile number	updating profile changingmobile num	1.change fist name2. change	you must enter to the dash board	a prompt message called "succe	a prompt message called "spass	
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	UP006	update profile	update password	1.insert password2.insert new password		a prompt message called "pass	a prompt message called "spass	
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	UP007	Dashboard button click	check dash button is working proper	1. press dashboard button		redirect to dash button	redirect to dash button	pass
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	UP008	Logout button	press logout button	1. press logout button		logout from account	logout from account	pass
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	UP009	mobile number validation	mobile number validation	1. enter text into mobile number field		prompt an message called "enter	prompt an message called "pass	
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	UP010	update password message	update password message	1. enter password 2. press update password message3. press b		prompt an message called "mess	prompt an message called "pass	
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	UP011	user profile interface style	user profile interface style	1. Press user profile		interface with good styles	page bottom have tytle issue	fail
Faculty Management	FM001	Select Faculty Management(left side nav)	selecting the sub title called " Faculty	1. Press user profile	you must enter to the dash board	display the faculty management	display the faculty manager	pass
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	FM002	Add new faculty management member wif	Add new faculty management memb	1.ignore first name field2. fill	you must enter to the dash board	Prompt and error message called	Prompt and error message	pass
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	FM003	Add new faculty management member wif	Add new faculty management memb	1.fill first name field2. ignore	you must enter to the dash board	Prompt and error message called	Prompt and error message	pass
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	FM004	Add new faculty management member wif	Add new faculty management memb	1.fill first name field2. fill last	you must enter to the dash board	Prompt and error message called	Prompt and error message	pass
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	FM005	Add new faculty management member wif	Add new faculty management memb	1.fill first name field2. fill last	you must enter to the dash board	Prompt and error message called	Prompt and error message	pass
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	FM006	Add new faculty management member wif	Add new faculty management memb	1.fill first name field2. fill last	you must enter to the dash board	Prompt and error message called	Prompt and error message called "first name	
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	FM007	Add new faculty management member wif	Add new faculty management memb	1.fill first name field2. fill last	you must enter to the dash board	Prompt and error message called	Prompt and error message	pass
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	FM008	Add new faculty management member wif	Add new faculty management memb	1.fill first name field2. fill last	you must enter to the dash board	add sucessfully	add sucessfully	pass
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	FM009	Add new faculty management member us	Add new faculty management memb	1.fill first name field2. fill last	you must enter to the dash board	Prompt and error message called	Prompt and error message	pass
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	FM010	Add new faculty management member wif	Add new faculty management memb	1. ignore all field2. press new	you must enter to the dash board	Prompt and error messages call	Prompt and error messages	pass
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	FM011	check validation of email field	check validation of email field	1. add incorrect format email address		display an error message called	display an error message	pass
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	FM004	search faculty member using faculty name	search faculty member using faculty	1. select faculty name		filter the information related to fa	filter the information related	pass
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	FM004	search faculty member using first name	search faculty member using first na	1. fill first name		filter the information related to fir	filter the information related	pass
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	FM004	search faculty member using last name	search faculty member using last na	1. fill last name		filter the information related to las	filter the information related	pass
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	FM012	view action	view action of member	1. press the view button of m	in the membe info. table. these	view the information of member	view the information of mem	pass
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	FM013	suspend action	suspend the membership of member	1. press the suspend button	in the membe info. table. these	suspend the member with succes	suspend the member with s	pass
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	FM014	add new faculty	add new faculty	1.. add university name2. add faculty name3. press add new facu		added new faculty successfully	couldn't add new faculty	fail
company manag	CM001	Select company management (left side na	selecting the sub title called "compan	1. Press company management		redirect to company managem	redirect to company manag	pass
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	CM002	Search organization by name	check whether Search organization	1. enter organization nameinto organization name field.		filter records with organization n	didn't filter	
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	CM003	Search organization by location	check whether Search organization	1. enter organization locationinto location field.		filter records with organization lo	didn't filter	
	Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
	CM004	display organization info	display organization info	1. press company management button		display all organization info	display all organization info	

2.46 JobsDB development

JobsDB is a CV generation site, which helps to generate CVS's for students freely.

It provides many cv templates and users can select one of them according to their interest.

Main advantage of JobsDB compared to other cv generation websites is, it is freely available and users can see their cv updating real time. Most cv generating websites don't have that feature. Because of that the user has to wait until the final phase to see their cv output.

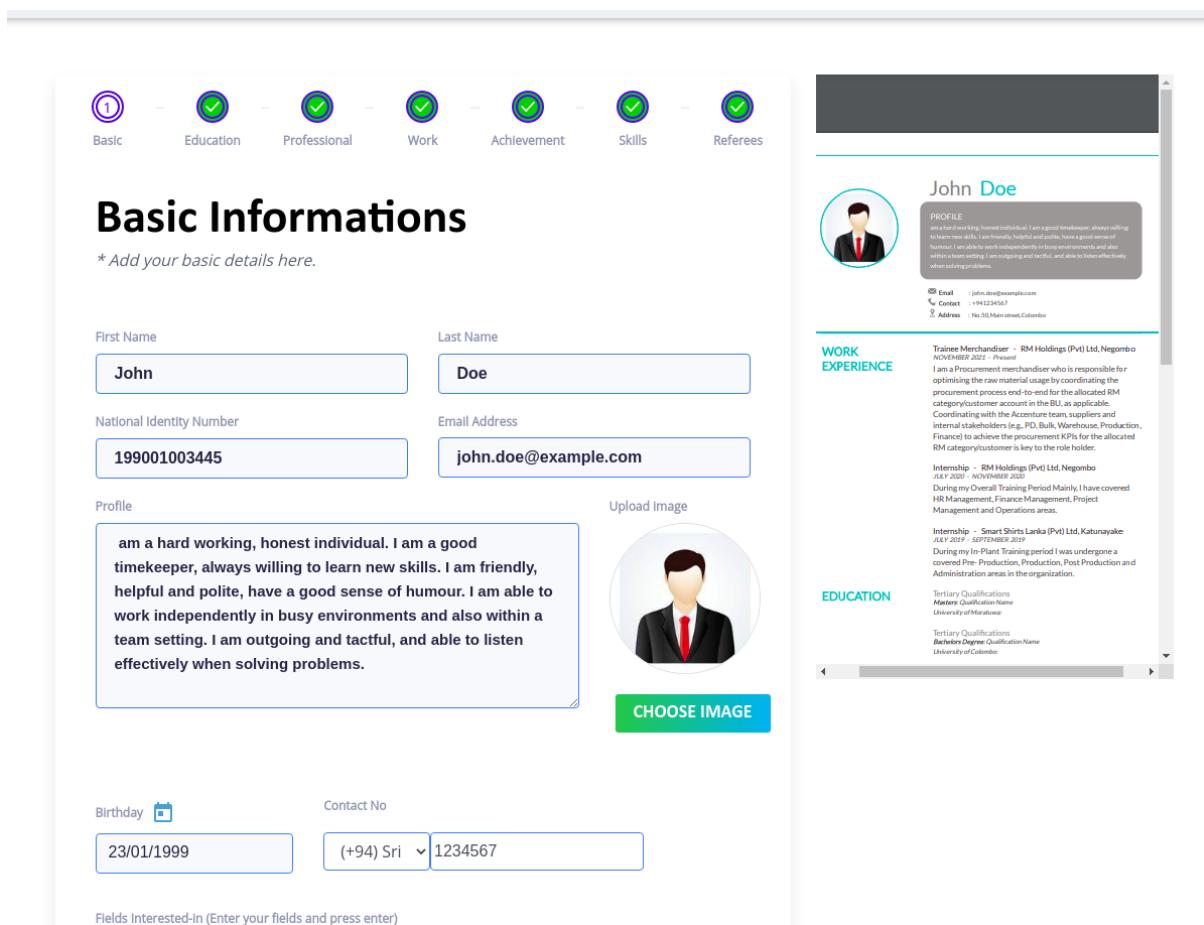


Figure 222:cv generation UI

I have been assigned to the JobsDB project as the cv developer.

Mainly I had two responsibilities as cv developer.

1. Update all CVS's, which are already available on the website.
2. Create new CVS's

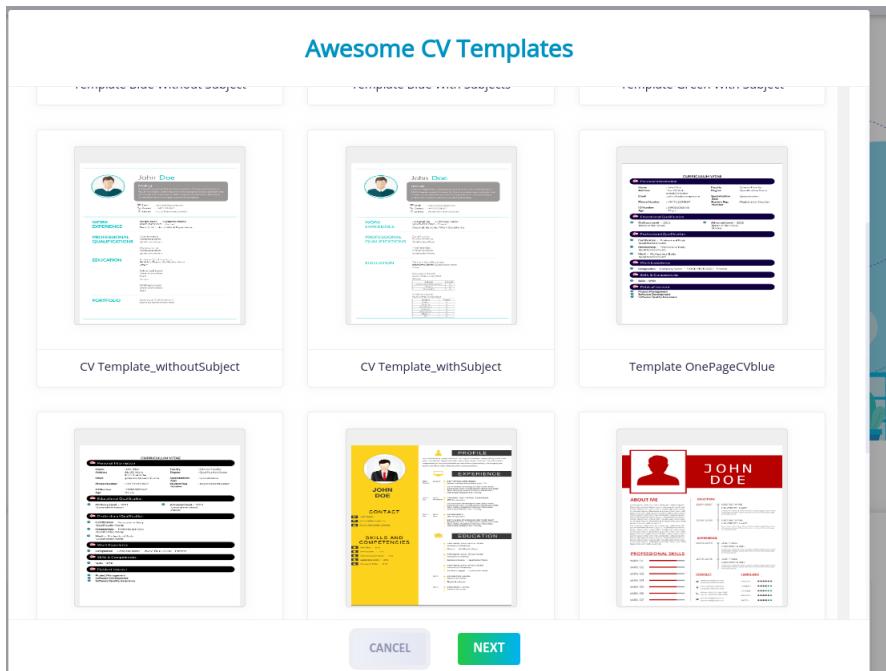


Figure 223:Available templates

According to the above image, first 4 CVS's were already available on the website and I was the person who updated them with some new logics. The last two CVS's developed by me from scratch.

All of 6 CVS's are already tested and issues fixed with many testing iterations.

From this level, I will describe each cv with their qualities and what I have done in each cv.

CV Template_withoutSubject



CV Template_withoutSubject

Figure 224:cv template without subject

In this cv, the supervisor asked me to fix the align issues and user Image position change as the first task.

After that task finished, the supervisor asked me to fix logic issues one by one as follows.

JobsDB templates consist of many sections as follows.

1. Basic Information
2. Educational Qualifications

Tertiary Qualifications: // User can enter many Tertiary Qualifications

G.C.E. Advanced Level: // User only can enter one Advanced Level

G.C.E. Ordinary Level: // User only can enter one Ordinary Level

Note: This template doesn't contain subjects.

3. Professional Qualifications

Note: Users can add many Professional Qualifications into this email. Some templates have limitations.

4. Work Experience

Note: Users can add many Work Experiences into this email. Some templates have limitations.

5. Other Achievements

Note: Users can add many Other Achievements into this email. Some templates have limitations.

6. Skills and Competencies

Note: Users can add many Skills and Competencies into this email. Some templates have limitations.

7. Non-Related Referees

Note: Users can add many Skills and Competencies into this email. Some templates don't contain Non-Related Referees sections.

Users can ignore any of each section according to his/her desires.

If a user ignores one of the sections from above, Its title cannot display in the template.

To handle this kind of logic issues, we have to use pebble template conditions.

It is a program language, which used to handle HTML logical issues.

But templates, which were available on the website in JobsDB, didn't handle this issue.

They assigned me to handle this issue and fix this issue.

I have added pebble conditions for each section to handle logic issues and completed the task within 3 days.

Next task was to handle the page break issue in Blue Template without subject.

“Blue Template without subject” doesn't have a page limit. users can add unlimited information for each section.

But when page breaks, we have to handle the page break properly.

Templates, which were added to the website, didn't handle page break issues, and the supervisor asked me to handle the page break issue.

The supervisor advised me to not use complex methods to handle the issue and asked me to research to find simple logic to handle the issue.

Because when templates get complex, hard to reuse or update by another developer.

I found a way to handle the issue using styles.

It was very simple. But it also had some limitations when applied to the templates.

But i decided to use it with some modifications to the HTML code.

```
style='page-break-inside: avoid;  
page-break-before:auto; display: block;
```

But I decided to use the above style within each section's section tag.

Then page break inside will not happen. It will be ignored by the above style.

After handling all of the logic issues.

I have to test the template within the website. There was a (test bed) staging website which is very similar to the live website, which used to test the changes under construction.

<https://jobsdb.hsenidjapan.hsenid.lk/home>

as the cv developer, I didn't have the permission to access the back end of the jobsdb.

But the supervisor gives me the admin permission to access the backend end points through postman requests.

He gave me the necessary permission and backend URL to post the postman request.

Through postman requests, I was able to add new templates and Delete existing templates.

First of all, I had to convert my cv code into json request format.

If there are icons or images in the template, I have to convert them to base64 or SVG format.

Because json requests don't support .jpg, .jpeg image formats.

If json requests a cv template ready to post, first we have to get the new token.

To get a new token, I have to have admin credentials.

After adding the token into the Authorization section, will be able to send the post request successfully.

If we are able to send a post request successfully, we will get the message "cv_name is successfully added" in the message body.

Following is a json format pebble template, which ready to post through postman.

```
{
  "name" : "CV Template_withoutSubject",
  "allowedRoles" : [
    "ROLE_STANDARD_USER"
  ],
  "hyperTextMarkup" : "<!DOCTYPE html><html><head><link rel='stylesheet' href='plnkr.css' /></head><body ><div style='page-break-after: always !important;' class='page-header' style='text-align: center'></div><div style=' page-break-after: always !important;' class='page-footer'></div><table><thead><tr><td><!--place holder for the fixed-position header--><div style=' page-break-after: always !important;' class='page-header-space'></div></td></tr></thead><tbody><tr><td><!--*** CONTENT GOES HERE ***--><div style=' page-break-after: always !important;' class='page'><div class='mainDetails'><div class='headerDiv1'><div id='headshot' class='quickFade'><img style='width: 4.0cm; height:4.0cm; margin-left:15px; position: relative' src='{{ bio.profileImageBase64 }}' alt='profile_pic' /><h2 class='job-title quickFade delayThree'></h2></div></div><div id='name' class='headerDiv2'><h1 class='quickFade delayTwo'><!-- {{ bio.firstName }} {{ bio.lastName }} --><span style='color: gray; font-weight: normal; text-align: right; margin-left: 75px; margin-top: -50%; position: relative'>&ampnbsp{{ bio.firstName }}&ampnbsp&ampnbsp</span><span style='color: #03b9b9; font-weight: normal; text-align: right;'>{{ bio.lastName }}</span><br /></h1>{{% if bio.profile is empty %}} {{% else %}}<div style='background: #8e8989e0; color: white; padding: 20px; border-radius: 1em; margin-top: 10px; margin-left: 65px; margin-right: 10px'><label>PROFILE</label><br /><span style='font-size:12px; margin-right:10px'>{{ bio.profile }}</span></div>{{% endif %}}</div><div class='headerDiv3'><div id='contactDetails' class='quickFade delayFour'><ul>{{% if bio.email is empty %}} {{% else %}}<li><span class='contact-icon' style='margin-left: 80px'><svg version='1.1' id='IconsRepoEditor' xmlns='http://www.w3.org/2000/svg' xmlns:xlink='http://www.w3.org/1999/xlink' x='0px' y='0px' width='16px' height='16px' viewBox='0 0 469.2 469.2' style='enable-background:new 0 0 469.2 469.2;' xml:space='preserve' fill='#808080' stroke='#808080' stroke-width='0'><g id='IconsRepo_bgCarrier'></g><path d='M22.202,77.023C25.888,75.657,29.832,74.8,34,74.8h401.2c4.168,0,8.112,0.85

```


Followings are the ids of all templates, which were available in JobsDB website.

If we want to delete one of them, we have to get id respect to the template, which needs to delete. And sent a postman request with POST.

The screenshot shows the Postman application interface. At the top, it displays 'Overview' and a 'POST' request to 'https://jobsdb.hsenidjapan.hsenid.lk/api/cv/administration/templates/delete/5fb7792ba5483a2f9f25...'. Below the request URL, the method is set to 'POST'. The 'Body' tab is selected, showing a single parameter '6a24'. Other tabs include 'Params', 'Authorization', 'Headers (7)', 'Pre-request Script', 'Tests', 'Settings', 'Cookies', and 'Beautify'. The 'Body' section has options for 'none', 'form-data', 'x-www-form-urlencoded', 'raw', 'binary', and 'GraphQL', with 'raw' selected. A large text area for the raw body content is present, with the value '1' entered. At the bottom, there is a 'Response' section.

Figure 226: Deleting existing templates

CV Template_withSubject



CV Template_withSubject

Figure 227:cv template with subject

cv Template_withsubject is very similar to the above template. The only difference is that it contains A/L and O/L subjects with table format.

I had the same issue fixing this template.

First of all, i fixed the CSS issues link update align and profile picture position changes.

After that task, I fixed the logic issues such as section page break issues and section title hide if used ignores that section.



John Doe

PROFILE

I am a hardworking, honest individual. I am a good timekeeper, always willing to learn new skills. I am friendly, helpful and polite, have a good sense of humour. I am able to work independently in busy environments and also within a team setting. I am outgoing and tactful, and able to listen effectively when solving problems.

Email : john.doe@example.com

Contact : +941234567

Address : No.50, Main street, Colombo

WORK EXPERIENCE

Trainee Merchandiser - RM Holdings (Pvt) Ltd, Negombo
NOVEMBER 2021 - Present

I am a Procurement merchandiser who is responsible for optimising the raw material usage by coordinating the procurement process end-to-end for the allocated RM category/customer account in the BU, as applicable. Coordinating with the Accenture team, suppliers and internal stakeholders (e.g., PD, Bulk, Warehouse, Production, Finance) to achieve the procurement KPIs for the allocated RM category/customer is key to the role holder.

Internship - RM Holdings (Pvt) Ltd, Negombo
JULY 2020 - NOVEMBER 2020

During my Overall Training Period Mainly, I have covered HR Management, Finance Management, Project Management and Operations areas.

Internship - Smart Shirts Lanka (Pvt) Ltd, Katunayake
JULY 2019 - SEPTEMBER 2019

During my In-Plant Training period I was undergone a covered Pre- Production, Production, Post Production and Administration areas in the organization.

EDUCATION

Tertiary Qualifications
Masters: Qualification Name
University of Moratuwa:

Tertiary Qualifications
Bachelors Degree: Qualification Name
University of Colombo:

Figure 228: cv blue without subject full cv

Peradeniya Template



Template OnePageCVblue

Figure 229:Peradeniya template

Peradeniya template is a special template designed for Peradeniya University.

It was a request from university and designed according to their requirements.

It is a simple one-page cv template and doesn't contain referee information and is limited to one or two records for each section.

This template was available with two colours. Navy blue and Black

But the source code is similar in both CVS's.

As above templates. First of all, I was asked to fix CSS issues in the template.

Because of many CSS issues. I decided to redesign this template from scratch and apply the pebble conditions to this template.

Not only the section title issue, I had to write a logic to handle the record limitations only.

Ex: If a user tried to add more than 2 work experiences into a template. I have to ignore the 3rd work experience for this template. Because this template is limited to two work experiences.

I added the logic limitations according to Peradeniya university's requirements.

And there is no page break issue handled in this template. Because this template is limited to one page. But limited to one page.

CURRICULUM VITAE

Personal Information

Name	: John Doe	Faculty	: Science Faculty
Address	: No.50, Main street, Colombo	Degree	: Qualification Name
Email	: john.doe@example.com		
Phone Number	: +94 1234567		
ID Number	: 199001003445		
Age	: 22 yrs		

Educational Qualification

-  Ordinary Level - 2015
Name fo the school
-  Advanced Level - 2018
Name of the school
Physical science

Work Experience

-  Trainee Merchandiser - RM Holdings (Pvt) Ltd, Negombo - NOVEMBER 2021 - Present
-  Internship - RM Holdings (Pvt) Ltd, Negombo - JULY 2020 - 2020

Skills & Competencies

-  MS Office : 8/10
-  SAP System : 7/10
-  COmmunication Skills : 7/10
-  Leadership SKills : 8/10

Fields of Interest

-  Project Management
-  Human Resource Management
-  Supply chain & Logistics
-  Buisness Admnistration

Figure 230:Peradeniya blue template

Yellow Two Page CV



YellowTwoPageCV

Figure 231: Yellow template

The “Yellow Two_page” template was designed by myself from scratch.



JOHN DOE

CONTACT

+941234567
 john.doe@example.com
 No.50, Main street, Colombo

SKILLS AND COMPETENCIES

- MS Office : 8/10
- SAP System : 7/10
- Communication Skills : 7/10
- Leadership Skills : 8/10
- Teamwork Skills : 8/10



PROFILE

I am a hard working, honest individual. I am a good timekeeper, always willing to learn new skills. I am friendly, helpful and polite, have a good sense of humour. I am able to work independently in busy environments and also within a team setting. I am outgoing and tactful, and able to listen effectively when solving problems.



EXPERIENCE

2020 - january	present	 SOFTWARE ENGINEER hSend Software International (pvt) LTD <small>List in reverse chronological order (most recent experiences first); include position details and dates. Work History Academic Positions Fellowships Internships Research and Training</small>
2019 - july	2019 december	 TRAINEE SOFTWARE ENGINEER ABC Incorporation <small>List in reverse chronological order (most recent experiences first); include position details and dates. Work History Academic Positions Fellowships Internships Research and Training</small>
2018 - july	2019 june	 INTERNSHIP ABC Incorporation <small>List in reverse chronological order (most recent experiences first); include position details and dates. Work History Academic Positions Fellowships Internships Research and Training</small>



EDUCATION

2018	 TERTIARY QUALIFICATIONS University of Moratuwa Masters : Qualification Name
2015	 TERTIARY QUALIFICATIONS University of Colombo Bachelors Degree : Qualification Name
	 TERTIARY QUALIFICATIONS University of Peradeniya Bachelors Degree : Qualification Name
	 ADVANCED LEVEL Name of the school Physical science
	 ORDINARY LEVEL Name fo the school

Figure 232:Yellow template

175

It was a two_page template which developed with limitation for each section

Also handled the page break issue with CSS and handled section title issue with pebble condition logics.

Within this template, I try to increase the adding beautiful CSS for template, even it is hard for json request formats.

2.47 Kitcut testing

Kitcut Japan is a circuit board printing company in Japan. Its head office is located in Osaka, Japan and circuit productions are done in China.



Figure 233:Kitcut Japan

hSenid Software International is the company, which is developing and maintaining the website of Kitcut Japan.

Through the following link, you can access the homepage of Kitcut Japan Company.

<https://kitcutlabo.com/>

Through their website, users can request to print their circuit designs as circuit boards with measurements.

User just have to upload his/her circuit designs with burger extension and they will send you the possibility of printing.

After the confirmation of payment, they will add that circuit to print and will send the circuits to your doorsteps via express mail.

I was the one who was responsible for testing of login and registration interface testing. Through the following link, you can access all the test cases that were written to test the login and registration.

https://docs.google.com/spreadsheets/d/12_s6sDaVoIwO0vMF4tz70jGG3rVxlMGC/edit#gid=511823217

Kitcut Login Interface

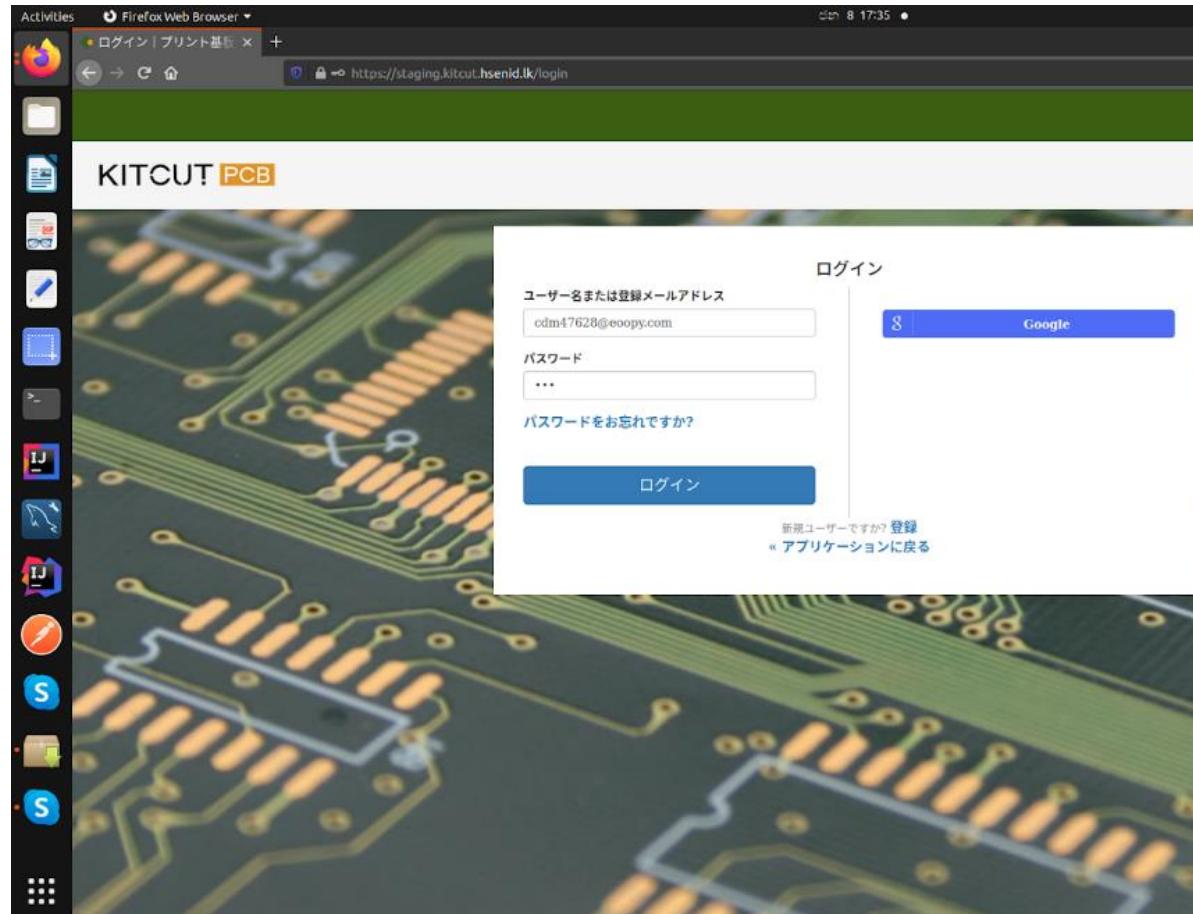


Figure 234:Kitcut login

Kitcut registration Interface

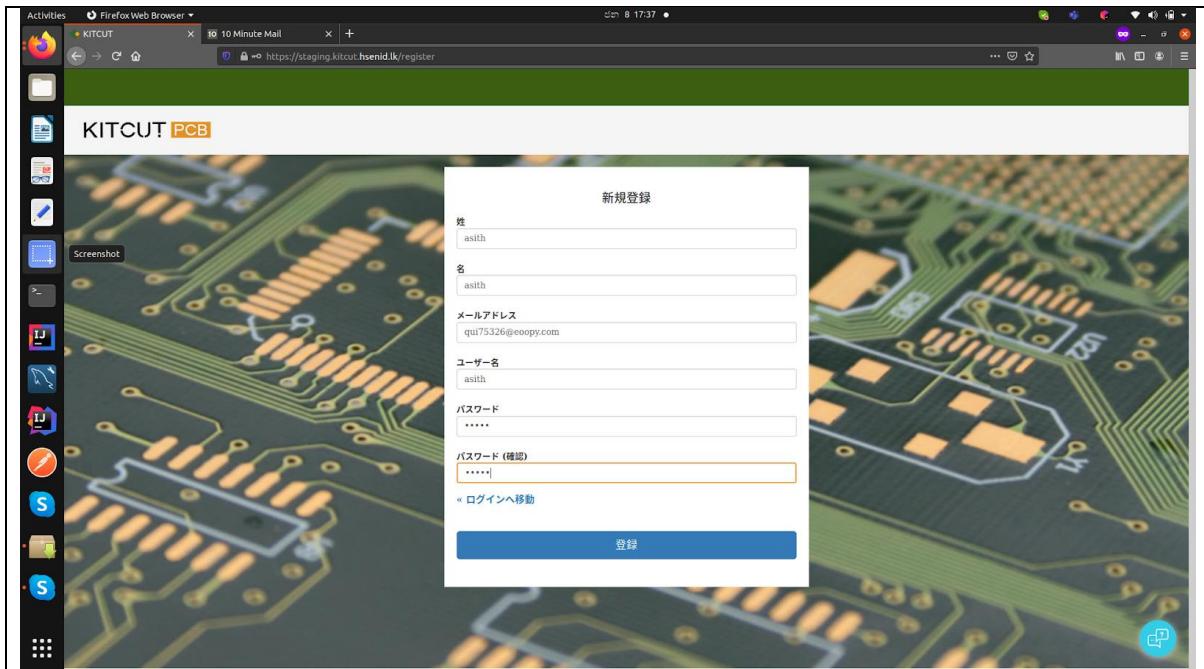


Figure 235:Kitcut registration

confirmation email sent message



Figure 236:confirmation email sent message

Confirmation email

Eメールの確認



service@kitcutlabo.com (service@kitcutlabo.com) just now
To: ebj32412@cuoly.com

このメールアドレスで asithish アカウントが作成されました。以下のリンクをクリックしてアカウント作成を完了してください。

[メールアドレスの確認](#)

このリンクの有効時間は 30時間です。

メールアドレスの確認を忘れた場合は、[TOPページ](#)から認証情報でログインしてください。その後、確認メールを再送信できます。

営業日 / 営業時間：（月曜日から金曜日） 10: 00 ~ 17: 00

土曜・日曜・祝祭日・弊社指定休日を除く

ご不明な点がございましたらお手数ですが、[こちらからお問い合わせください。](#)

Figure 237:Confirmation email

Kitcut Homepage

The screenshot shows the homepage of the KITCUT PCB website. On the left, there's a sidebar with various icons for file operations like Open, Save, Print, and Copy. The main header features the KITCUT PCB logo. Below the header, there's a large promotional banner for a "試作開発応援キャンペーン開催！" (Sample Development Support Campaign). The banner highlights an Amazon gift card worth 2,000 yen and mentions that 50 winners will receive it. It specifies the campaign period from October 1, 2020, to December 31, 2020. To the left of the banner, there's a "見積りスタート" (Estimate Start) button with input fields for board size (100 x 100 mm, 5 pieces), thickness (2層 (両面), 1.6mm), and a "見積りスタート！" (Start Estimate!) button. At the bottom of the page, the slogan "KIT × CUT × LABO" is displayed, along with links for "開発のためのプロトタイプ" (Prototyping for development), "[コスト]を削減する" (Reduce costs), and "研究室・実験室" (Research lab/experiment room). The footer contains the KITCUT PCB logo and a link to the login page (<https://staging.kitcut-hanboldt.jp/logout>).

Figure 238:Kitcut Homepage

Forget password interface – email type

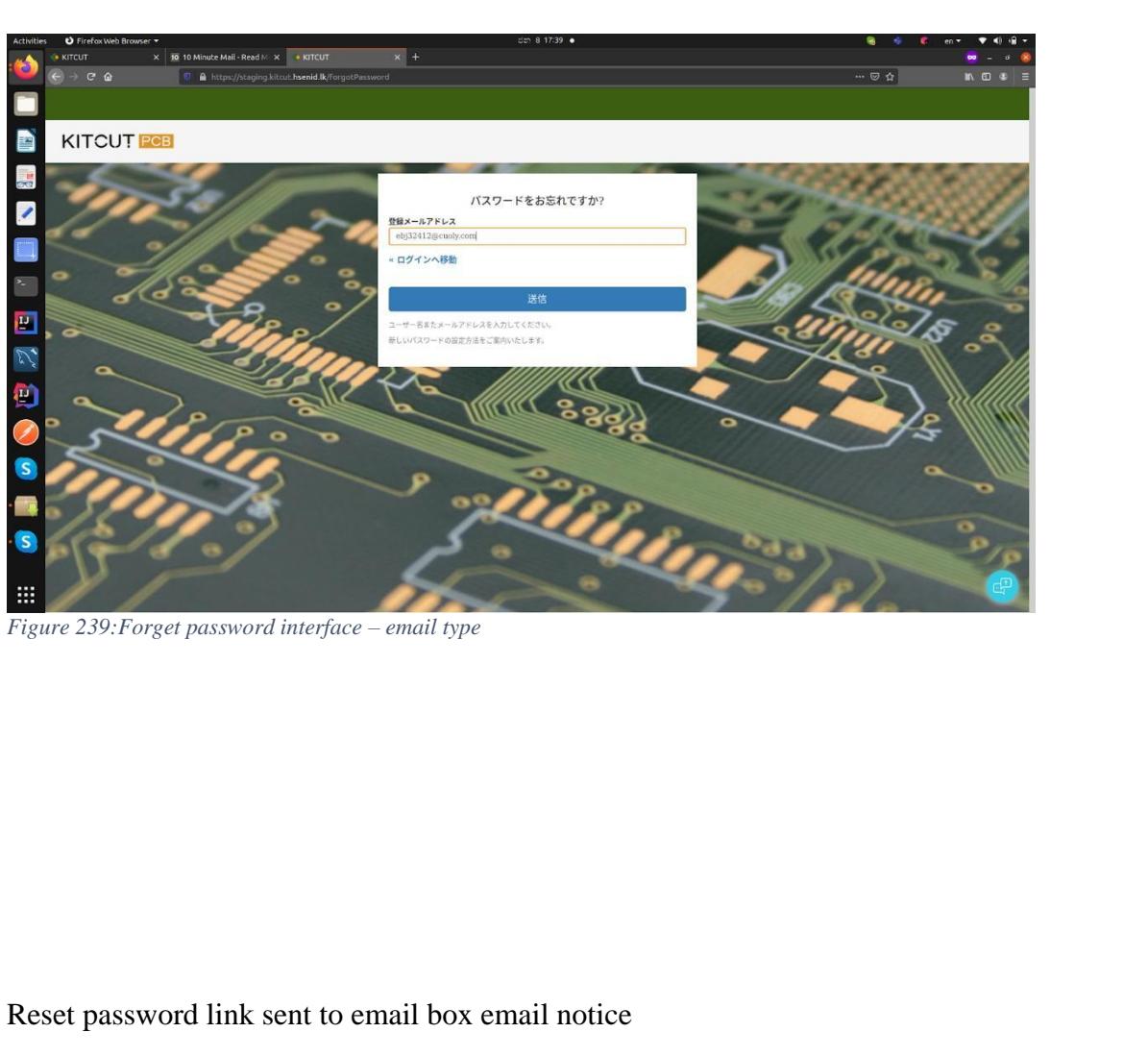


Figure 239:Forget password interface – email type

Reset password link sent to email box email notice

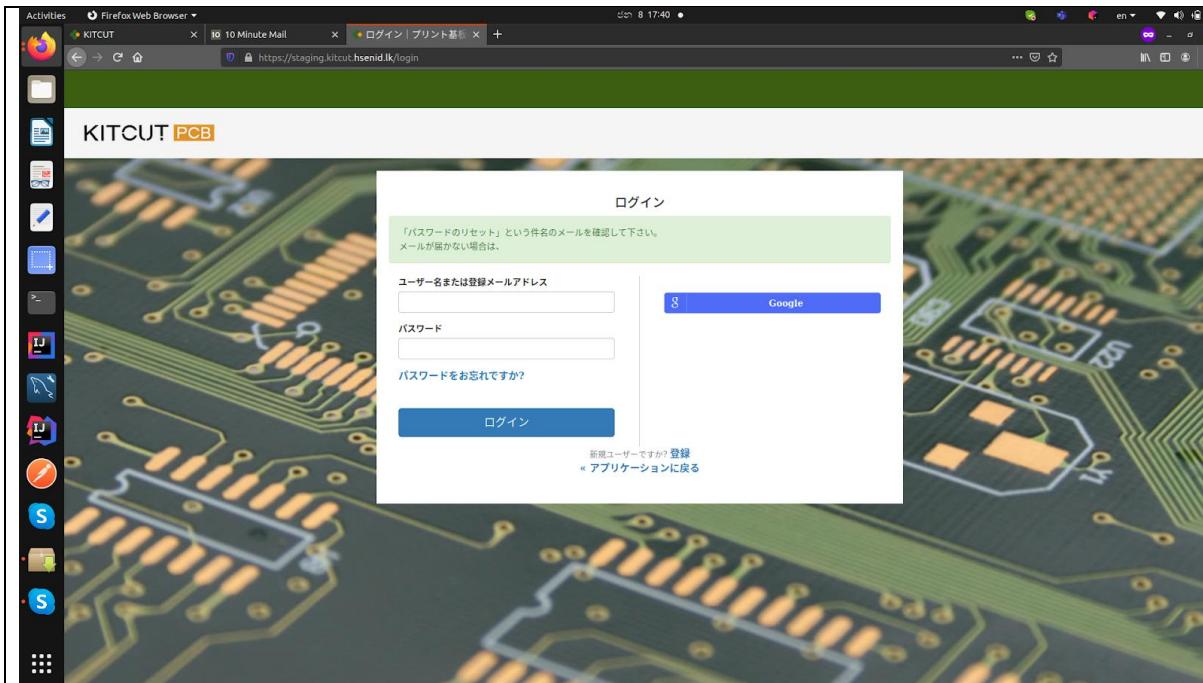


Figure 240:Reset password link sent to email box email notice

Password reset email

パスワードのリセット



service@kitcutlabo.com (service@kitcutlabo.com) just now

To: ebj32412@cuoly.com

あなたの asithish アカウントのパスワードの変更が要求されています。以下のリンクをクリックしてパスワードのリセットを行ってください。

[パスワードのリセット](#)

このリンクは 09-01-2021 08:10:08 だけ有効です。

もしパスワードのリセットを行わない場合は、このメッセージを無視してください。何も変更されません。

Figure 241:Password reset email

Enter new password and re-confirm password

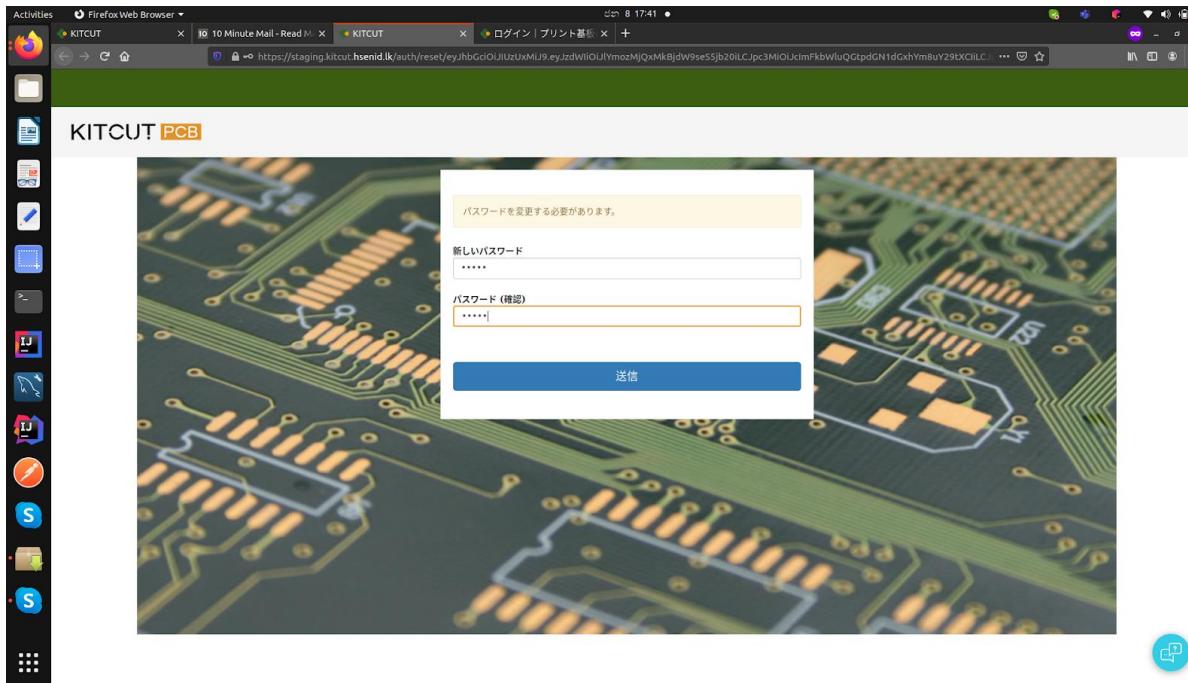


Figure 242: Enter new password and re-confirm password

Enter username box error message- textbox validation check

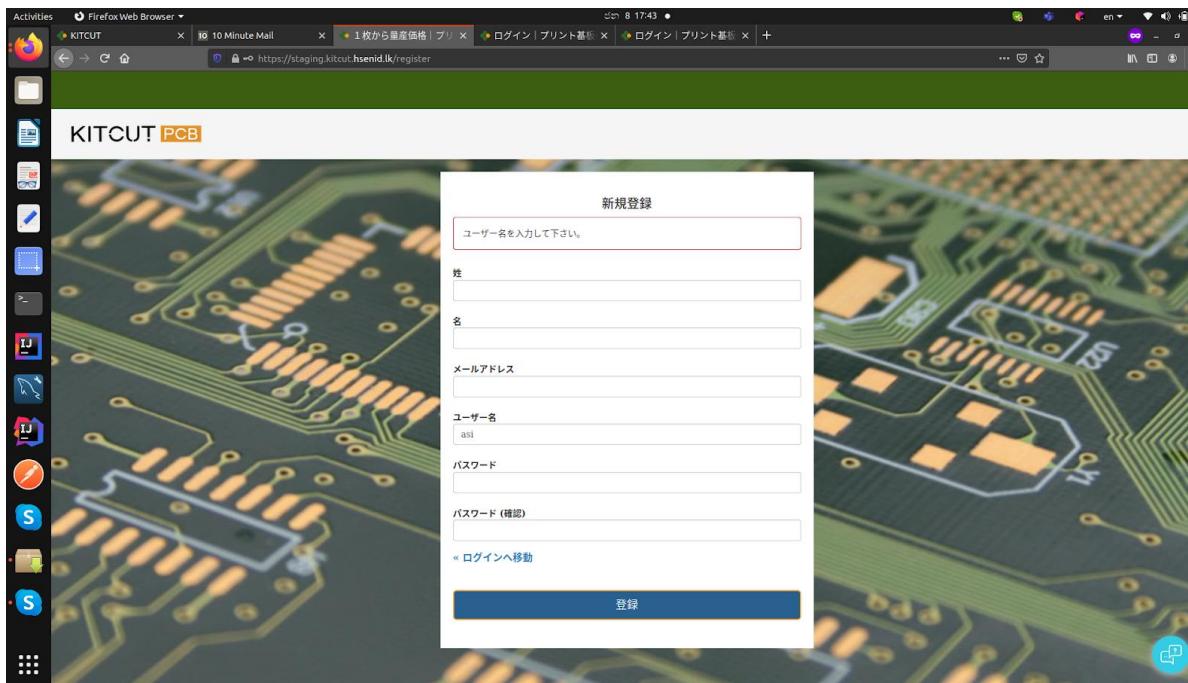


Figure 243:username box error message- textbox validation

Empty email box validation test

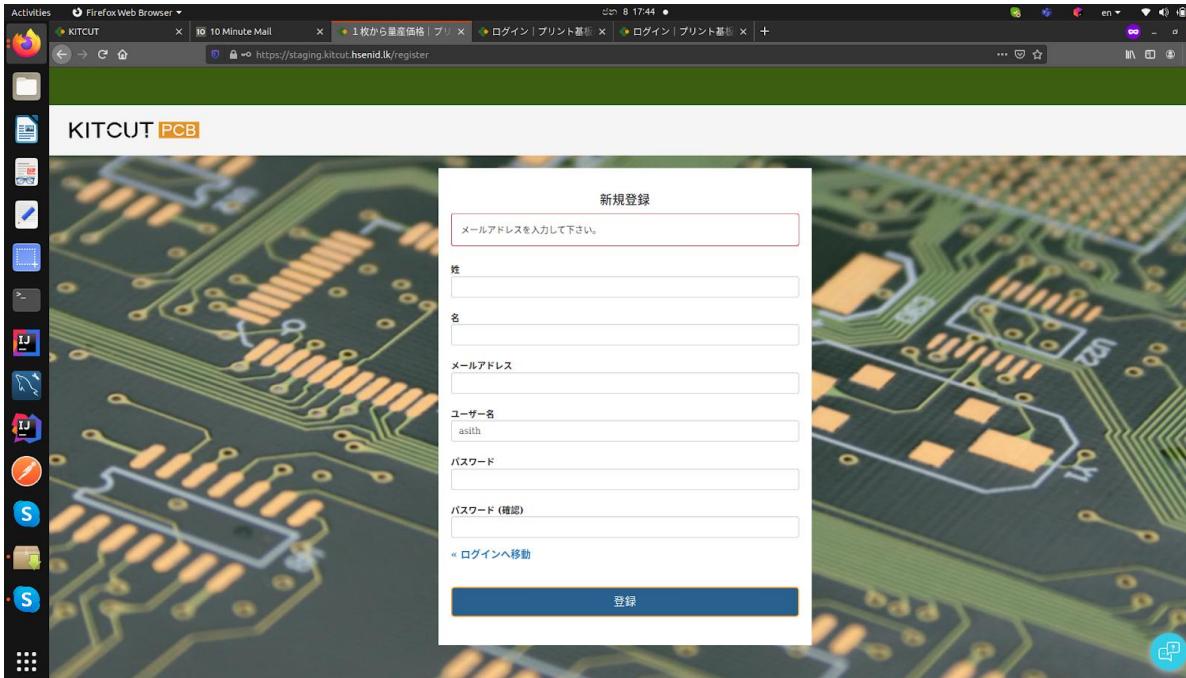


Figure 244: Empty email box validation test

Empty password box validation test

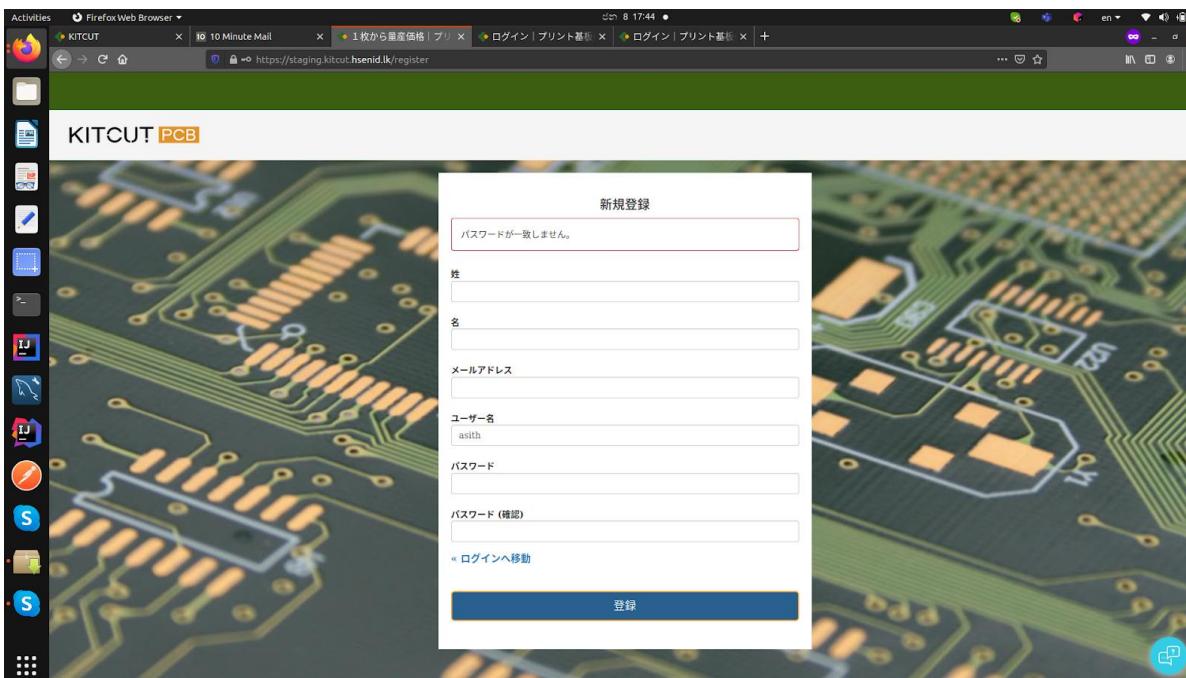


Figure 245: Empty password box validation test

Error message called- “username is already used”

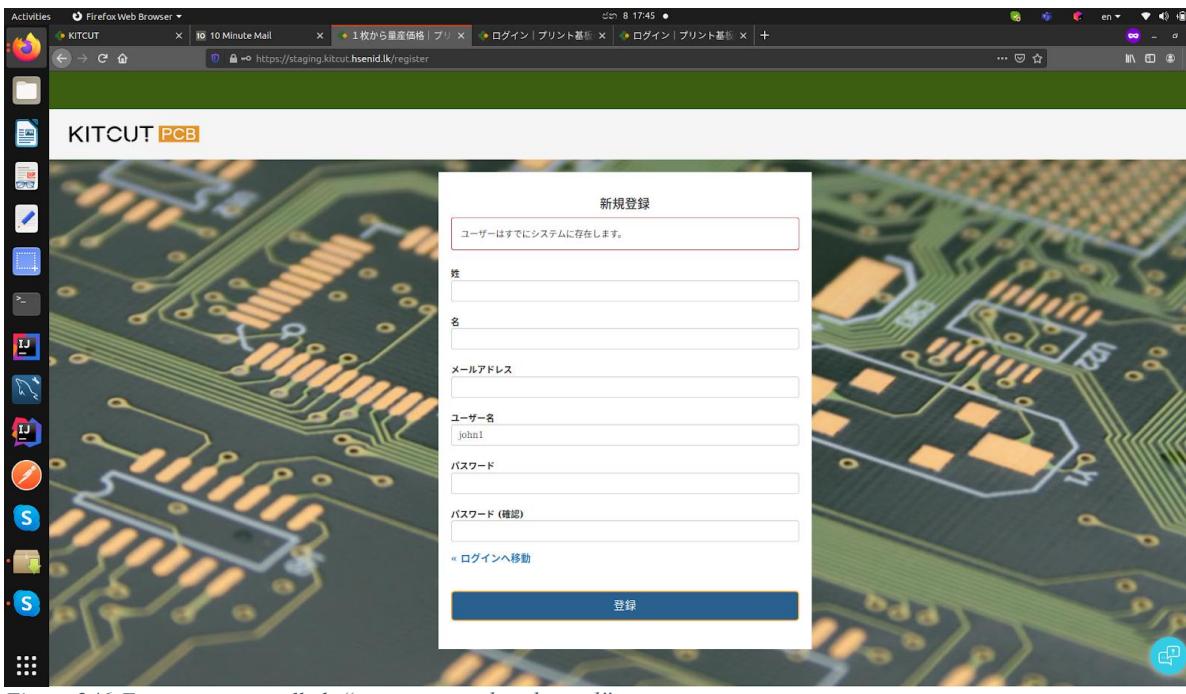


Figure 246: Error message called- “username is already used”

2.47.1 login testing

Test Case No	Test Title	Test summary	Test steps	Preconditions	results	Actual results	Status
L001	Login Page	Press Login			Quick loading of	Page loaded	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	results	Actual results	Status
L002	Login Page	Page Loading	1. Go to home	User must have	Quick loading of	Page loaded	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	results	Actual results	Status
L003	Enter Values to	Enter Valid	1. Enter Valid	User Must have	Prompted to	Prompted to	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	results	Actual results	Status
L004	Enter Values to	Enter Valid	1. Enter Valid	User Must have	Prompted to	Prompted to	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	results	Actual results	Status
L005	Enter Values to	Enter Invalid	1. Enter Valid	Password must	Display	Displayed Invalid	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	results	Actual results	Status
L006	Enter Values to	Enter Invalid	1. Enter Invalid	Email Address	Display	Displayed	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	results	Actual results	Status
L008	Forgot Password	Click Forgot	1.enter email	User Must have	Prompt to Forgot	Prompted to	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	results	Actual results	Status
L009	Forgot Password	Click Reset	1.Click Forgot	User Must have	Display	Received Reset	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	results	Actual results	Status
L009	check Forgot	Click Reset	1.Click Forgot	User Must have	receives rest	receives rest	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	results	Actual results	Status
L010	Recieve Forgot	Click Reset	1.Click Forgot	User Must have	User should	User received	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	results	Actual results	Status
L011	Reset Password	Click Reset	1.Click Reset	User Must have	Prompt to Reset	Prompted to	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	results	Actual results	Status
L012	Enter New	Enter New	1.Enter New	User Must have	1.Display	1.Displayed	Pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	results	Actual results	Status
L013	Enter New	Enter New	1.Enter New	User Must have	Display	Display	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	results	Actual results	Status
L014	Enter New	Enter New	1.Enter New	User Must have	Display Error	Display Error	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	results	Actual results	Status
L015	Enter New	Keep Fields	1. Keep	User Must have	Display Error	Display Error Me	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	results	Actual results	Status
L016	google login	check if google	1. go to login	user must have	redirect to home	redirect to home	pass

2.47.2 Registration Testing

Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R001	Register button click	press Register button	1. go to https://kicubobo.com:8180/ website 2. press register	internet connection is needed	load the registration page	registration page is loaded	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R002	Registration page load time	After the press the register button, rel1. press the register button		internet connection is needed	load the registration page within few seconds	registration page is loaded within few seconds	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R003	Input values to fields	enter the true values in the field	1.enter sur name2. enter the first name3. enter an email address	you must have an unregistered email	prompt a message to email validation	prompt an message to email validation	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R004	input true values except sur name	enter the true values in the field except sur name	1.ignore sur name2. enter the first name3. enter an email address	you must have an unregistered email	prompt surname is required	prompt surname is required	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R005	input true values except first name	enter the true values in the field except first name	1.enter sur name2. ignore the first name3. enter an email address	you must have an unregistered email	prompt first name is required	prompt first name is required	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R006	input true values except surname and input true values except surname all	input true values except surname and input true values except surname all	1.ignore surname2. ignore the first name3. enter an email address	you must have an unregistered email	prompt surname is required	prompt surname is required	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R007	input true values except first name	enter the true values in the field except first name	1.enter sur name2. ignore the first name3. enter an email address	you must have an unregistered email	prompt first name is required	prompt first name is required	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R008	input true values except username	input true values except user name	1.enter surname2. enter the first name3. enter an email address	you must have an unregistered email	prompt username is required	prompt username is required	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R009	input true values except password	input true values except password	1.enter surname2. enter the first name3. enter an email address	you must have an unregistered email	prompt password is required	prompt password is required	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R010	input true values ignoring email, pass	input true values ignoring email, pass	1.enter surname2. enter the first name3. ignore an email address	you must have an unregistered email	prompt email, password, re-password is required	prompt email, password, re-password is required	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R011	input true values ignoring first name	input true values ignoring first name	1. ignore surname2. ignore the first name3. ignore an email address	you must have an unregistered email	prompt firstname, surname and password is required	prompt firstname, surname and password is required	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R012	Register click button with empty field	Register click button with empty field	1. ignore all fields2. press register button in register form		prompt all fields fill is required	prompt all fields fill is required	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R013	press link to migrate to login page	when press login page migration link	1. ignore all fields2. press login page migration link		redirect to login page	redirect to login page	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R014	email format validation	when enter without valid format, to d1. fill all fields 2. fill email box without following valid email format3. press register button			prompt an error message called invalid email	prompt an error message called invalid email	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R015	Validating the email address with esq	Validating the email address with esq	1.enter text only name or text and numerics combination name with @sign		nothing prompt as an error	nothing prompt as an error	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R016	Matching the two passwords with val	make sure with valid info.	1. type password in password field2. type same password (re-password) in re-password field.		nothing prompt as an error.	nothing prompt as an error.	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R017	Matching the two passwords with un	Matching the two passwords with un	1. type password in password field2. type different password (re-password) in re-password field.		prompt an error message called "passwords do not match"	prompt an error message called "passwords do not match"	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R018	Cannot register with previously registered email	If someone use email for register in	1. fill all fields with necessary info2. fill email field with already used email address3. press the register button		prompt and error called "email already in use"	prompt and error called "email already in use"	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R019	Cannot register with previously registered username	If someone use username for register in	1. fill all fields with necessary info2. fill email field with already used username3. press the register button		prompt and error called "username already in use"	prompt and error called "username already in use"	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R020	"Email verification mail sent" message	If someone register with information	1. enter all information2. press register button.		after pressing the register button, prompting of	after pressing the register button, prompting of	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R021	Login without confirming the registration	If user try to login using not confirm1. login with unregistered email2. press register button.3.use email and password to login without confirmation			prompting message "please confirm your email"	prompting message "please confirm your email"	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R022	verification email receive time	checking verification email receive	1. after press registering2. go to email inbox3. wait for the email.		email will receive within 1 minute.	email is received within 1 minute.	pass
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R023	Password Strength measurement	check if there is a password strength	1. check enter password without case sensitive and symbols.		show the strength measurement	didn't show	fail
Test Case No	Test Title	Test summary	Test steps	Preconditions	Expected results	Actual results	Status
R024	verification email click	press verification email link to confirm	1. send confirmation email to the email inbox 2. visit email box	user must press the verification email link	after pressing the email verification, page will	after pressing the email verification, page will	pass

2.48 Final day of the Internship

In final day of my internship. I had a different task from before. TechLead asked me to do the knowledge transferring session for the other two employees.

Responsibilities what I had until last day had to transfer with some tutoring.

Also, I shared all resources and credentials what I had used until last day.

Chapter 3 - Conclusion

I was trained under hSenid Software Lanka (Pvt.) Ltd. The industrial training is a great opportunity for a software engineering undergraduate to get exposure to the Industry. It was a great opportunity for me to get knowledge of real projects in the software industry. The projects I contributed gave me huge knowledge on handling projects from the beginning to the end.

During the internship period, I realized that working in a real working environment with a real client is much more changeable than doing academic work in the university. The experience gained through this period increased the capability of working under pressure to meet deadlines, teamwork, knowledge sharing and especially converting theoretical knowledge to real world scenarios and applications. Also, by the internship, I got the opportunity to learn about time management, discipline, and effective communication skills, to become confident in my abilities, and to find my strengths and weaknesses. Since HSL treats its trainees and permanent employees in a similar way, I understood that I also have a responsibility.

Nevertheless, it is not just the technical experience that I got from the internship period at HSL but also experiences in an organizational environment gave me to improve my professional life and added some enjoyable time to my personal life. Furthermore, I learned about organizational policies, environment, culture and most importantly how to balance work with personal life.

References

- Beginners BashScripting.* (n.d.). Retrieved from
<https://help.ubuntu.com/community/BashScripting>
- beginnersbook.com.* (n.d.). Retrieved from <https://beginnersbook.com/2013/05/servlet-tutorial/>
- Hibernate Understanding.* (n.d.). Retrieved from
https://www.codeguru.com/cpp/data/mfc_database/sqlserver/article.php/c10079/Understanding-Hibernate-ORM-for-JavaJ2EE.htm#:~:text=It%20manages%20the%20database%20and,the%20database%20and%20the%20objects.&text=Hibernate%20generates%20SQL%20for%20you,delete%
- How To Install MySQL on Ubuntu 18.04.* (n.d.). Retrieved from
<https://www.digitalocean.com/community/tutorials/how-to-install-mysql-on-ubuntu-18-04>
- How to Install Tomcat 9 on Ubuntu 18.04.* (n.d.). Retrieved from
<https://linuxize.com/post/how-to-install-tomcat-9-on-ubuntu-18-04/>
- Java FX.* (n.d.). Retrieved from https://docs.google.com/document/d/1PtW-KcKy6MovwsDMFAjAZedB_WNPtxD5WLV3FXTdVRU/edit?usp=sharing
- JSP understanding.* (n.d.). Retrieved from
<https://docs.google.com/document/d/1MeBVbdihH8HXcFmcrZQb8iPhht-EFK77erRTdbn5g6A/edit?usp=sharing>
- Learning basic Git commands.* (n.d.). Retrieved from
https://docs.google.com/document/d/1zRNC9dXLDo29wxgXoV1bLwXSi6NNTrIJ8_pV7Sp3PI/edit?usp=sharing
- Spring MVC tutorial.* (n.d.). Retrieved from
[https://www.agilealliance.org/glossary/tdd/#q=~\(infinite~false~filters~\(postType~\(~'page~'post~'aa_book~'aa_event_session~'aa_experience_report~'aa_glossary~'aa_research_paper~'aa_video\)~tags~\(~'tdd\)\)~searchTerm~'~sort~false~sortDirection~'asc~page~1\)}](https://www.agilealliance.org/glossary/tdd/#q=~(infinite~false~filters~(postType~(~'page~'post~'aa_book~'aa_event_session~'aa_experience_report~'aa_glossary~'aa_research_paper~'aa_video)~tags~(~'tdd))~searchTerm~'~sort~false~sortDirection~'asc~page~1)})
- Spring MVC tutorial.* (n.d.). Retrieved from <https://www.javatpoint.com/spring-mvc-tutorial>
- spring_boot/spring_boot_introduction.* (n.d.). Retrieved from
https://www.tutorialspoint.com/spring_boot/spring_boot_introduction.htm
- Springboot build a RESTful web API.* (n.d.). Retrieved from <https://spring.io/guides/gs/rest-service/>
- Teaching/Unix.* (n.d.). Retrieved from <http://www.ee.surrey.ac.uk/Teaching/Unix/index.html>
- understanding-angular-and-creating-your-first-application.* (n.d.). Retrieved from
<https://javascript.plainenglish.io/understanding-angular-and-creating-your-first-application-4b81b666f7b4>

Unit Testing Junit. (n.d.). Retrieved from
<https://www.vogella.com/tutorials/JUnit/article.html>

www3.ntu.edu.sg. (n.d.). Retrieved from
<https://www3.ntu.edu.sg/home/ehchua/programming/java/JSPByExample.html>