

To be *Technical*

Or not to be

THAT is the question!



# The Questions We Ask..

**No Future of Exploratory  
(Manual) Testing?**



# The Questions We Ask..

## How to Learn Automation ?

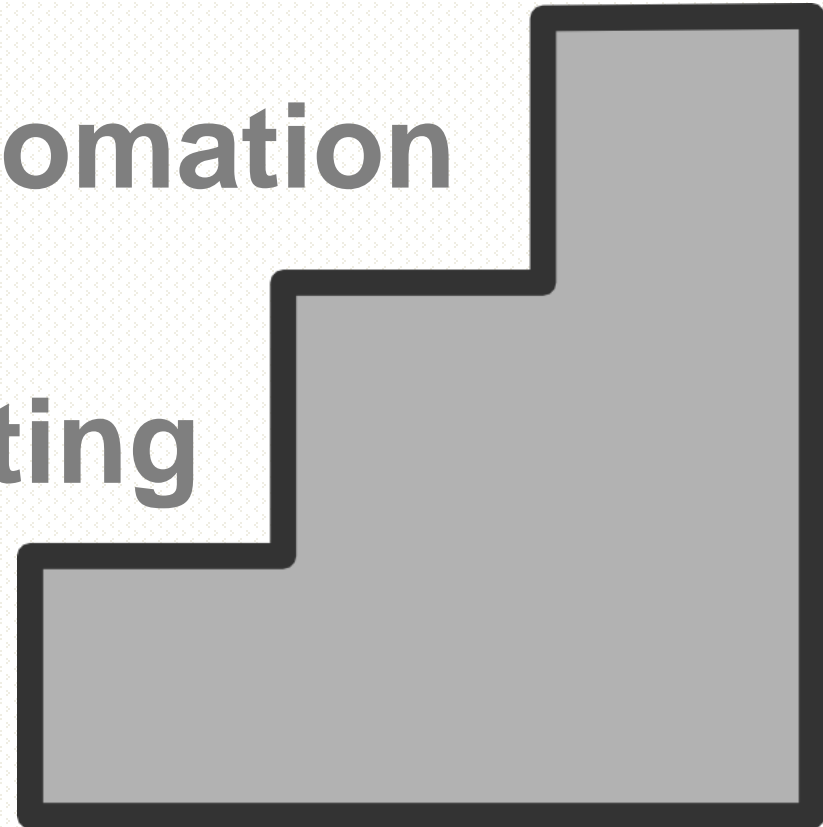


# The Perceived Career Path

**Management**

**Automation**

**Exploratory Testing**

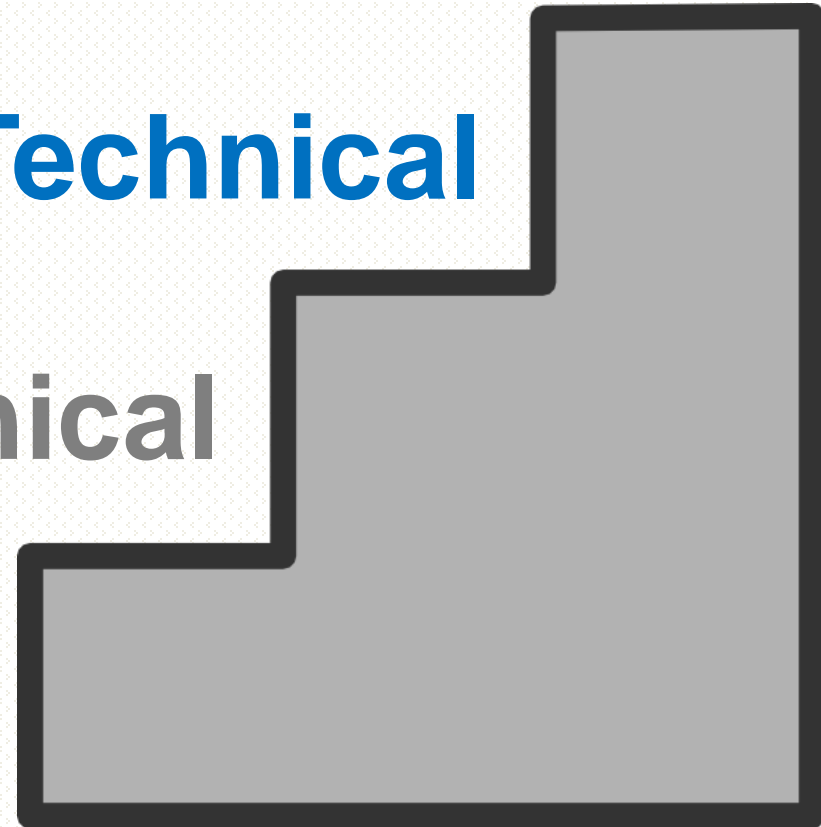


# The Actual Career Path

**Technical & Leadership**

**Technical**

**Non Technical**



# The Correct Question to Ask

To be **Technical**

Or *not* to be

**THAT is the Question!**



# About Me

**Ali Khalid**



AssetWORKS

**Redefining Software Quality**



**[www.Quality-Spectrum.com](http://www.Quality-Spectrum.com)**

# Meet Asim The Tester







# ASIM'S CAR TESTING STORY



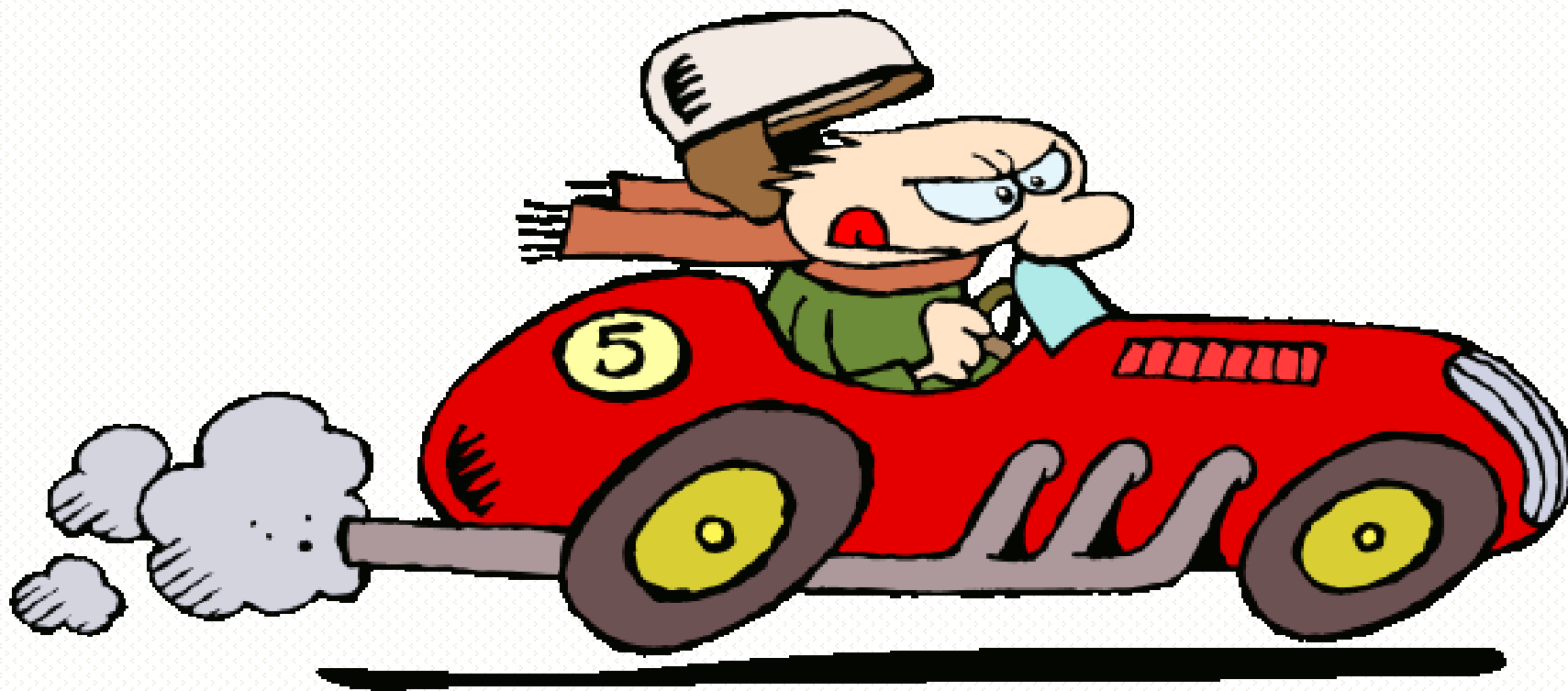
# Started Functional Testing



# Drive The Car



# At High Speeds



# Smash into the Wall



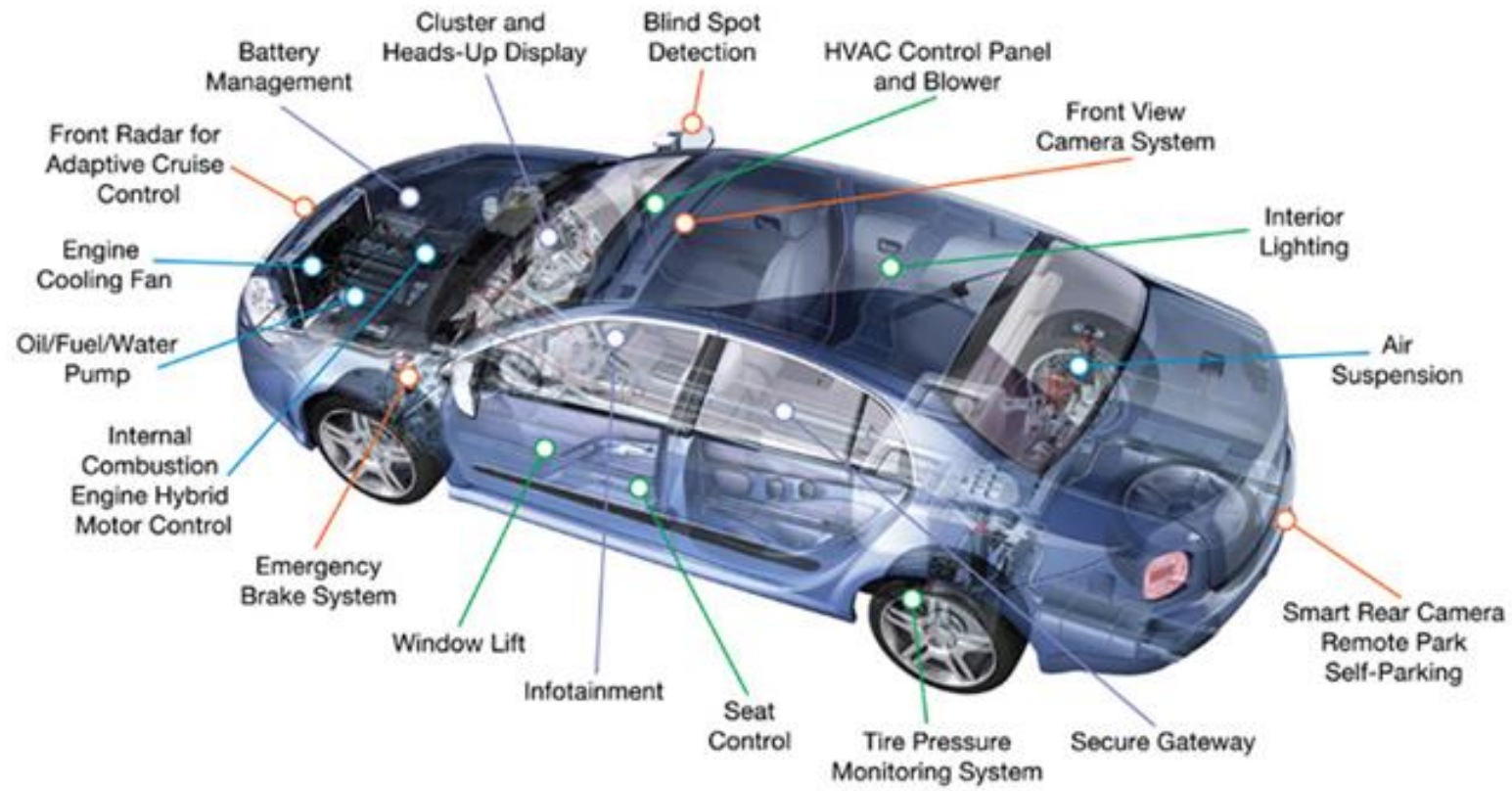
# Bean Style Testing



# Releasing to Production



# Untested Functionality in The Field





# Bugs Slipped Away

Toyota is recalling 2.1 million of its Prius, RAV4, Tacoma and Lexus vehicles globally because of a software problem that could cause cars to stop suddenly.



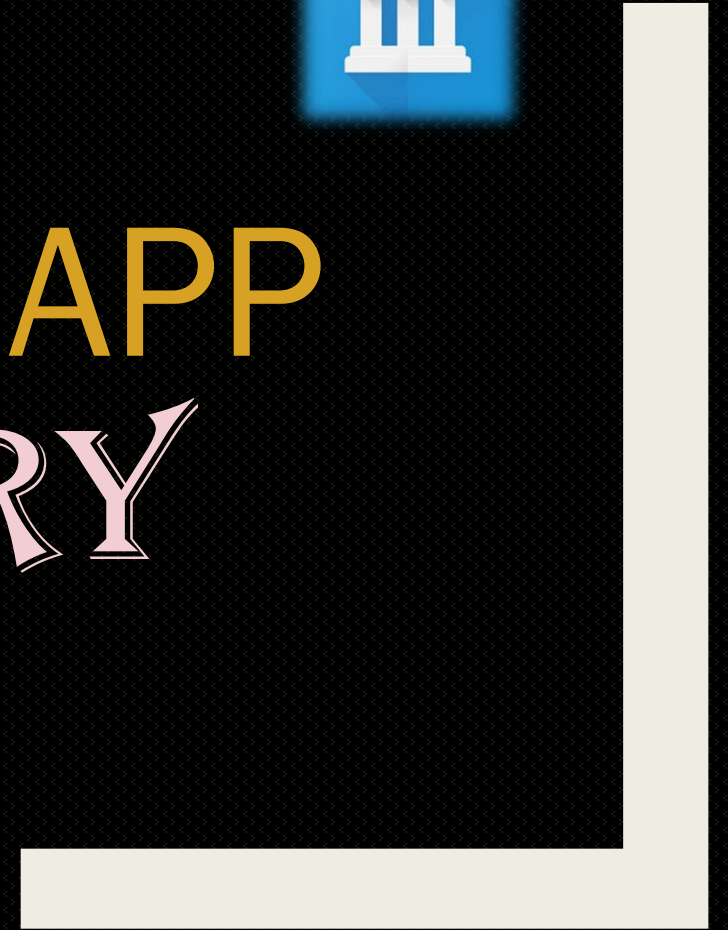
LESSON  
LEARNED

---

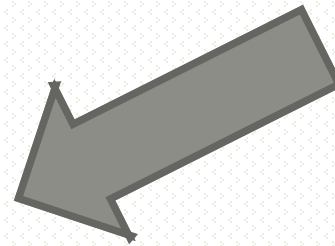
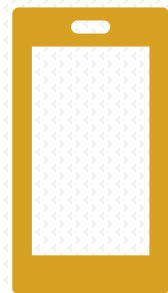
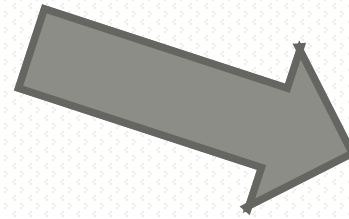
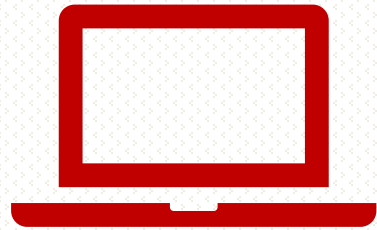
**Testers MUST BE**  
**Technical**



# ASIM'S HYBRID APP TESTING STORY



# The Tests



**Server**

# UI Testing

Front End

UI

Functional

UI Design

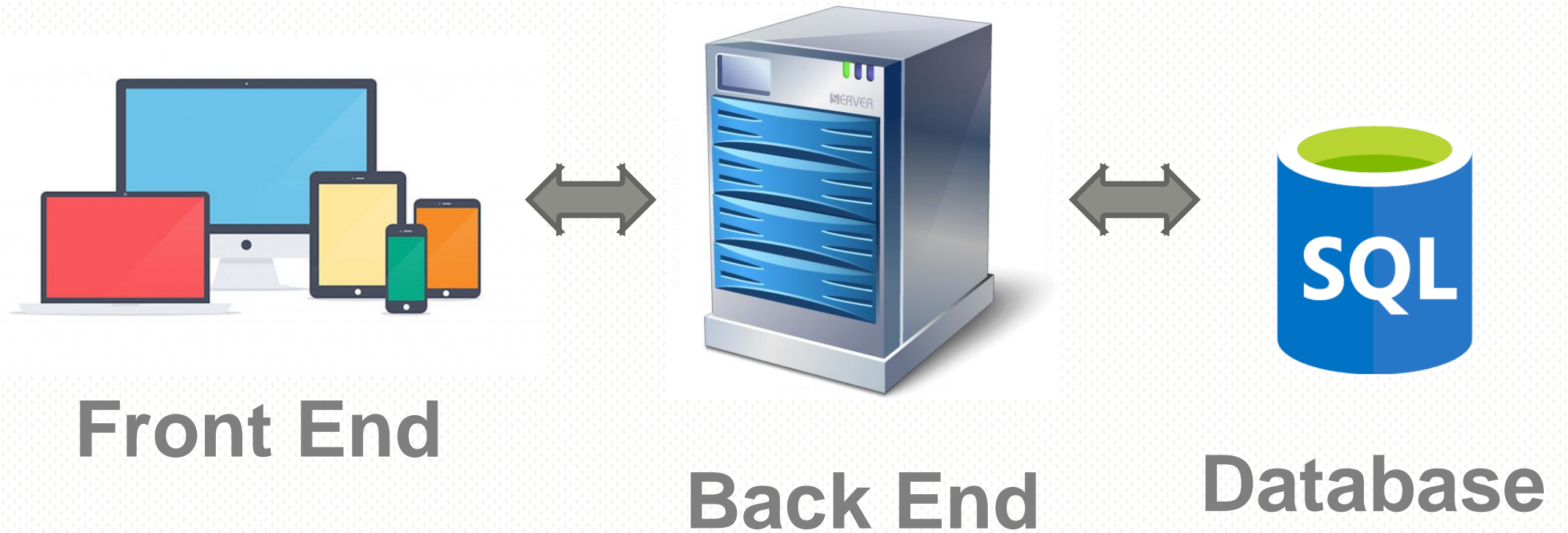
Compatibility



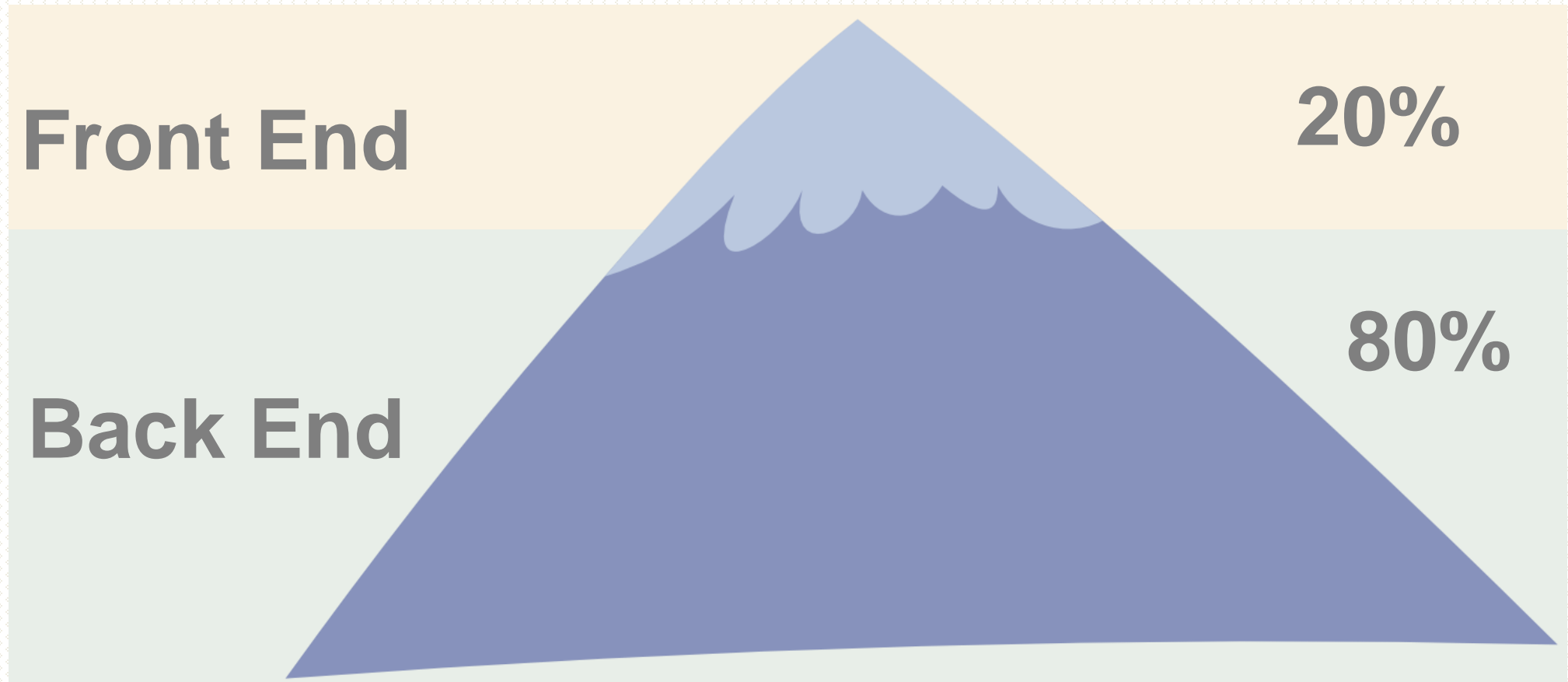
# How to Get Maximum Coverage?



# Understanding The Technology Stack

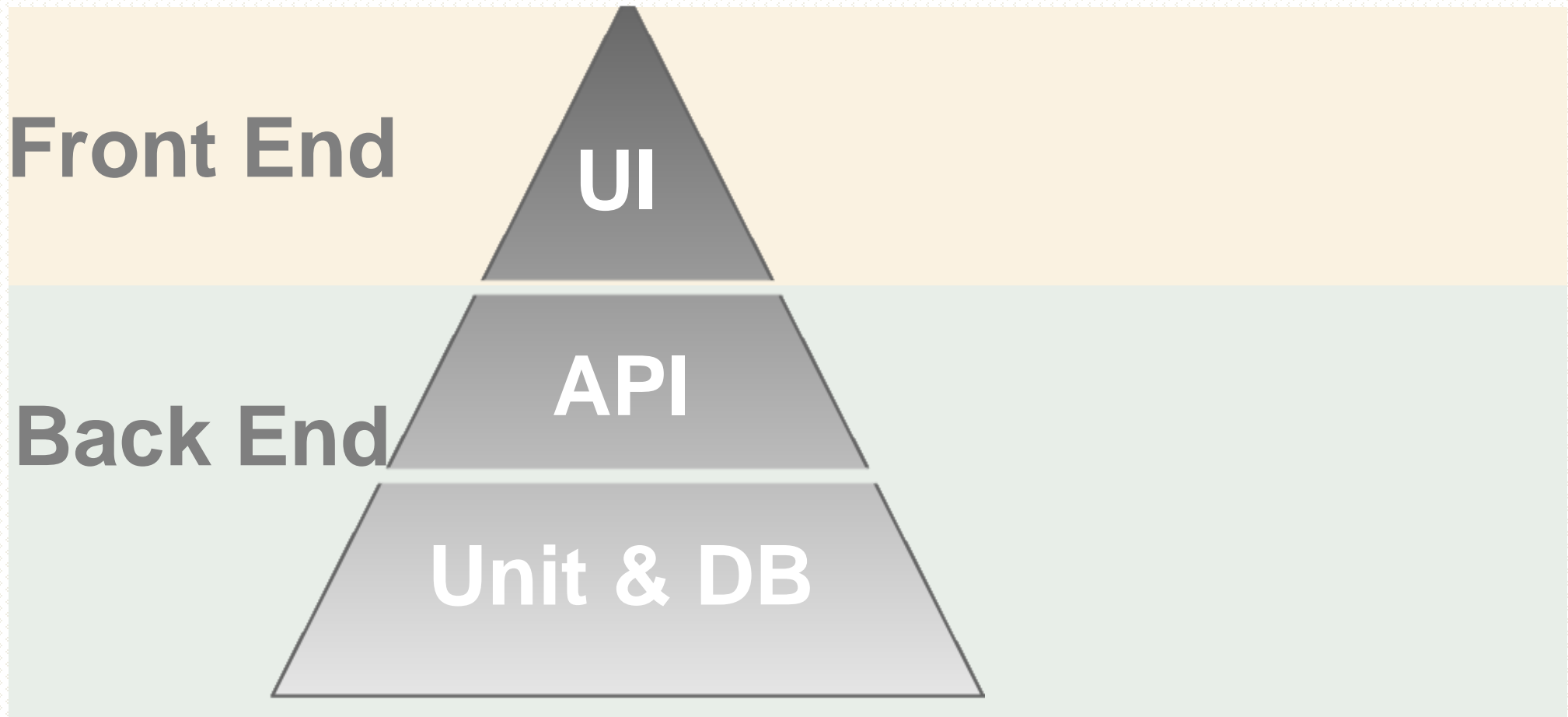


# Amount of Code

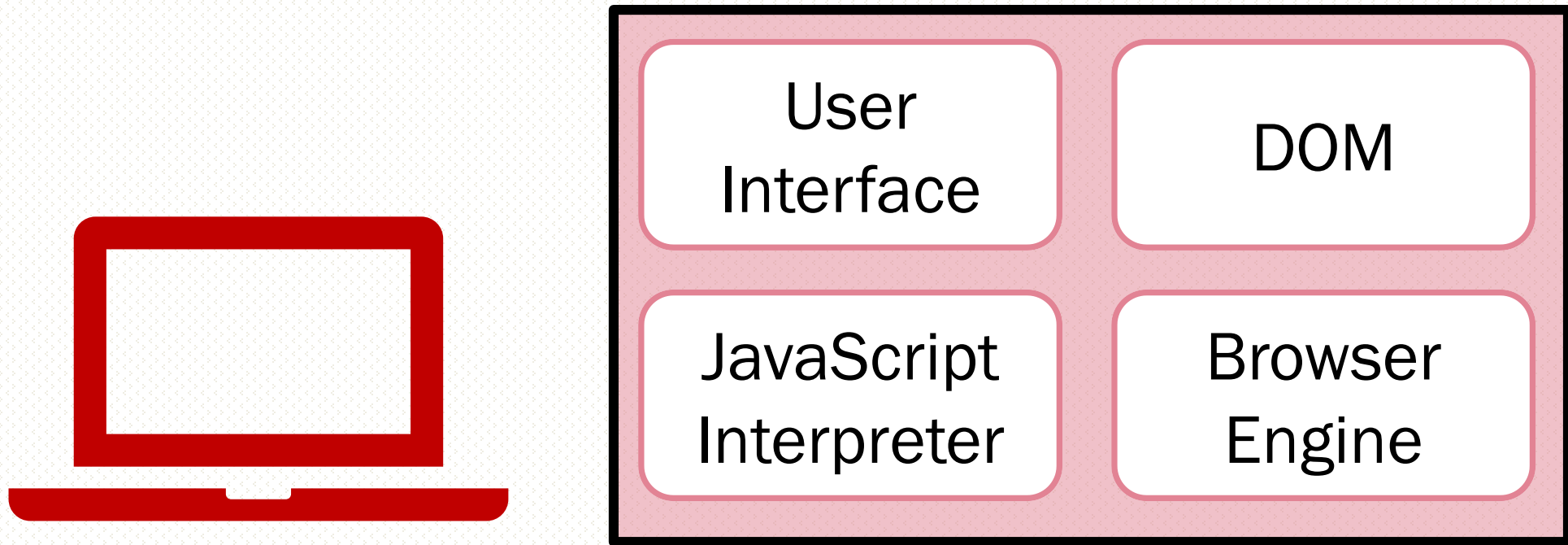




# Automation Pyramid

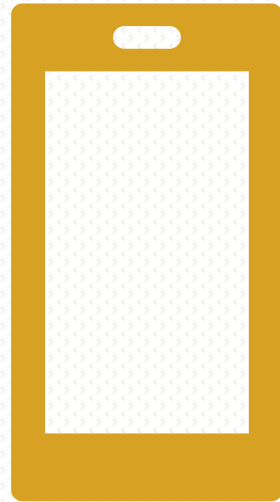


# Front End - Web

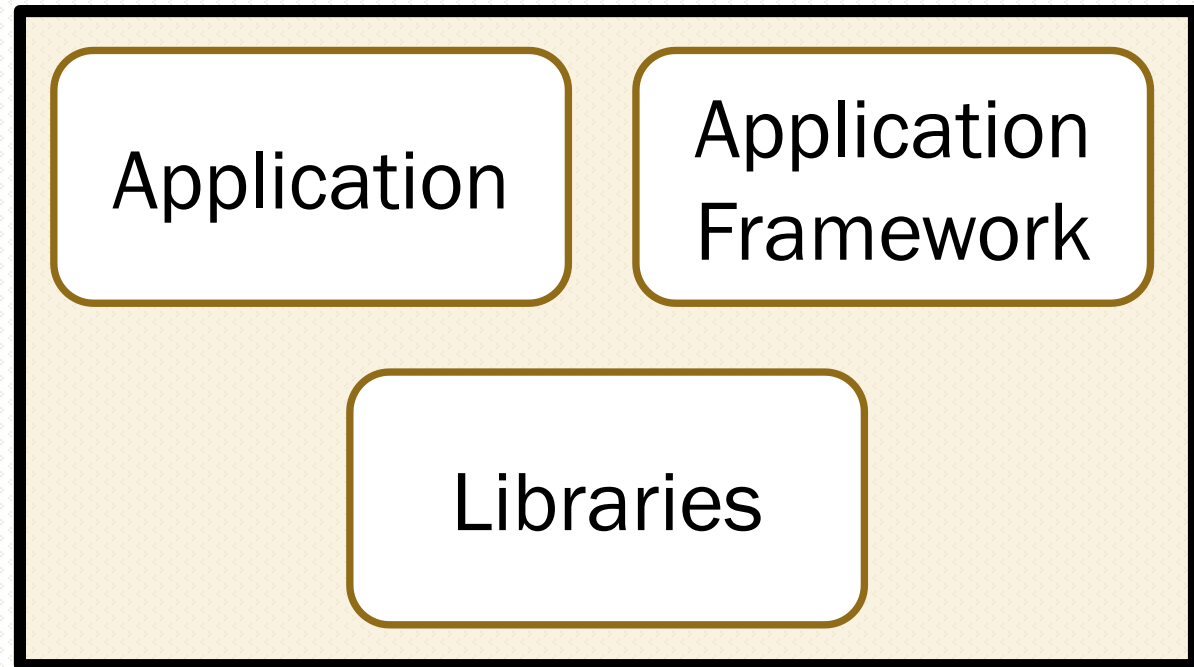


Browser  
Architecture

# Front End - Mobile



ANDROID



**Android  
Architecture**

# API's



Front End

{ api }

HTTP Request



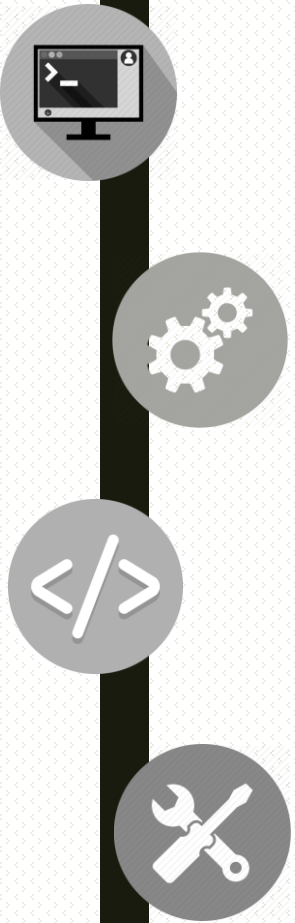
{ api }

HTTP Response

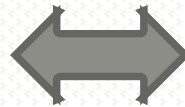


Back End

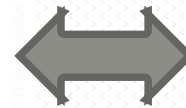
# Back End



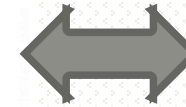
API



Web Server



Application Server



Database Server

# API Testing



**Back End**



**API**

**Module tests**

**Integration tests**

**Third party services**

# Unit & Database Tests



Data validation

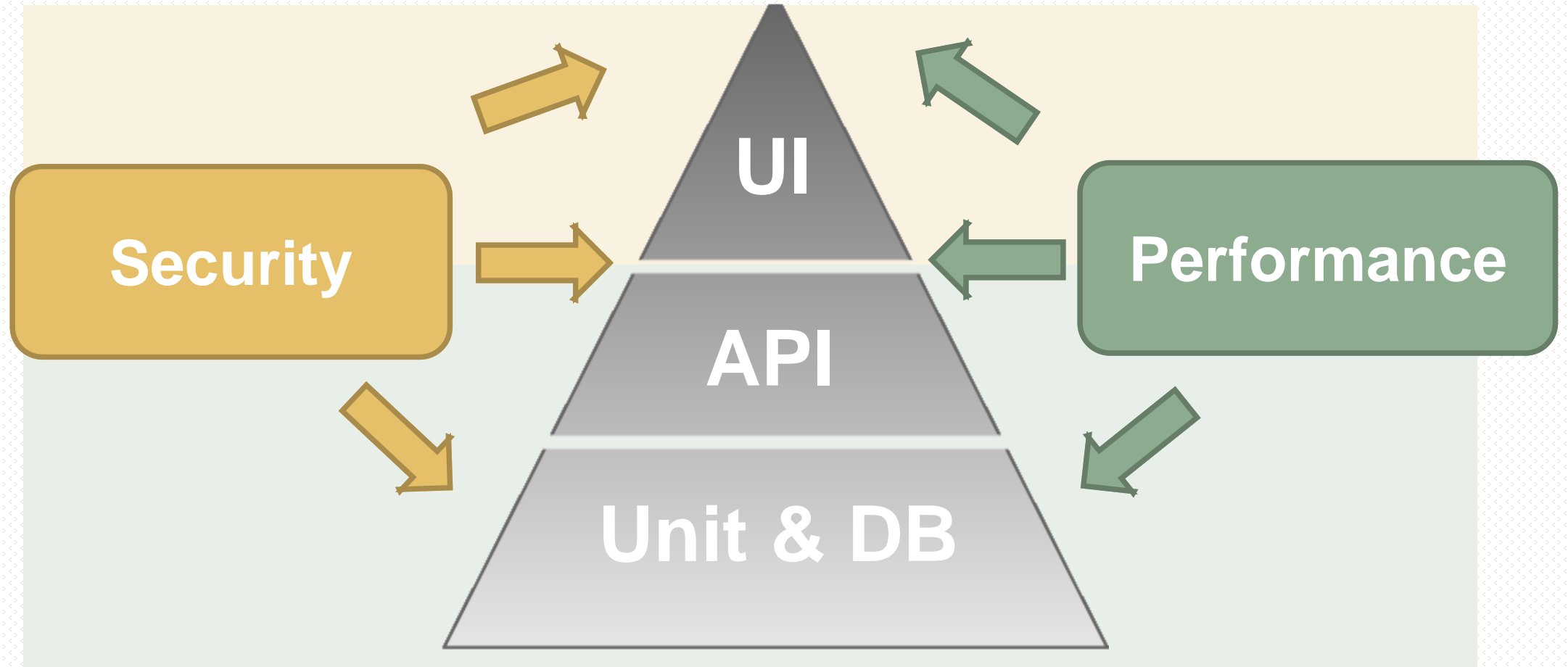
Data creation

Back End

Unit & DB

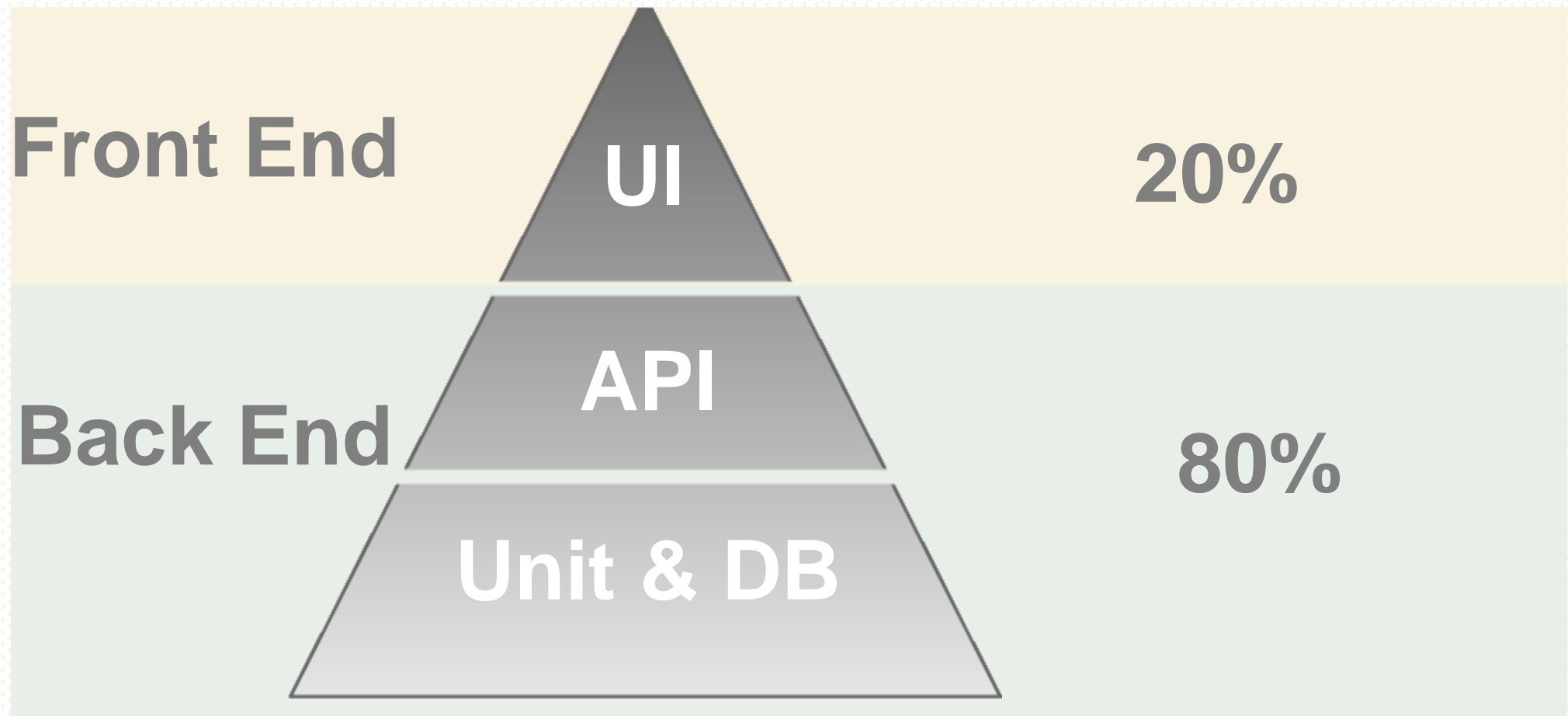
Unit tests

# Testing Across the Stack





# Automation Pyramid



LESSON  
LEARNED

---

**Test Across the  
Technology Stack**

# Technology Stack Examples – MEAN

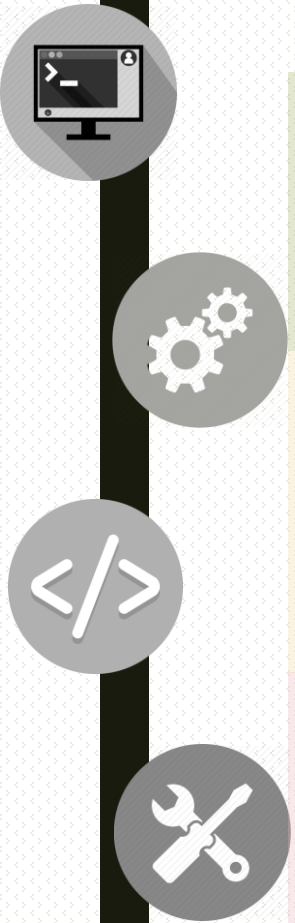
Front  
End



Back  
End



DB



# Technology Stack Examples – Microsoft

Front  
End

HTML



CSS



JS



Back  
End

ASP.NET | Web API

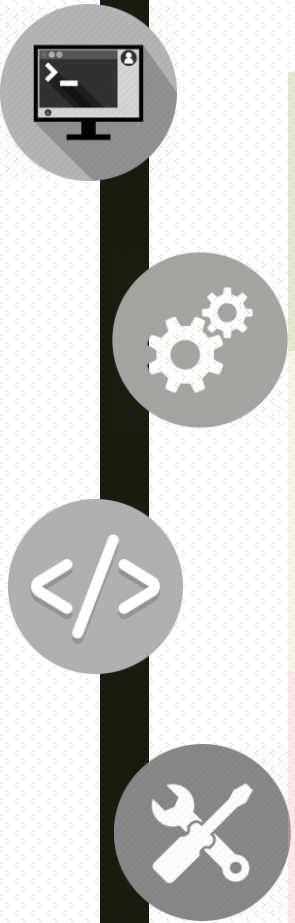
Microsoft  
ASP.net™

.NET MVC™

DB



Microsoft®  
SQL Server®



# Technology Stack Examples - JAVA

Front  
End



BACKBONE.JS



Back  
End



Java EE™



DB



# Technology Stack Examples

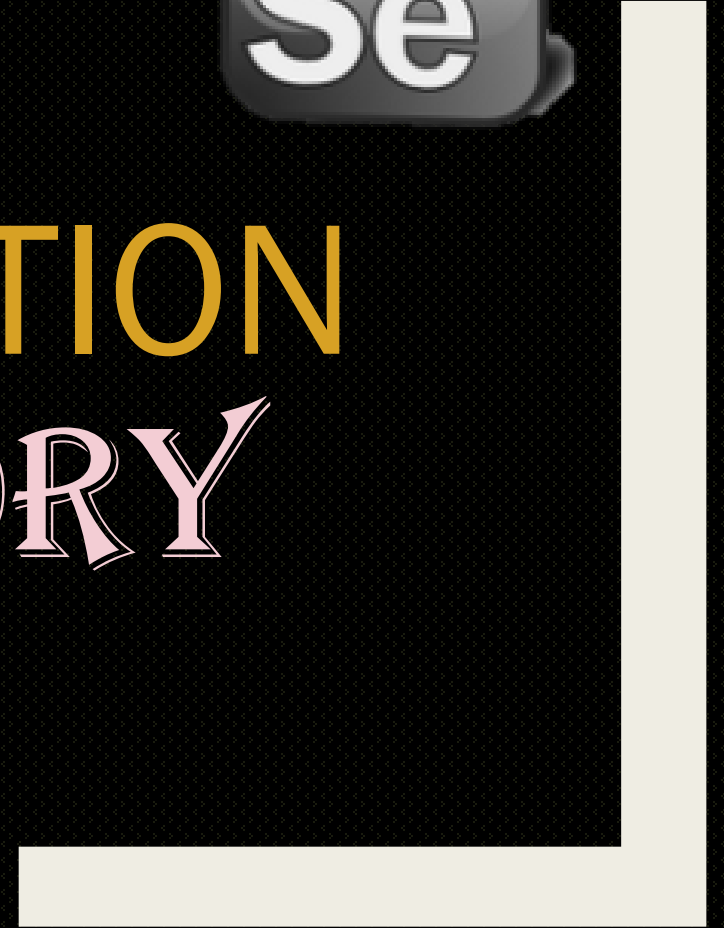


<https://stackshare.io/>





# ASIM'S **AUTOMATION** LEARNING *STORY*



# 15+ Years Coding Experience?



## Automation Engineer Position

- > 15+ experience in Java, Python and Ruby
- > Inventing the next version of Selenium
- > Experience trouble shooting Space X rockets





# Learned Two Dozen Tools?



## Automation Engineer Position

> Expertise on:

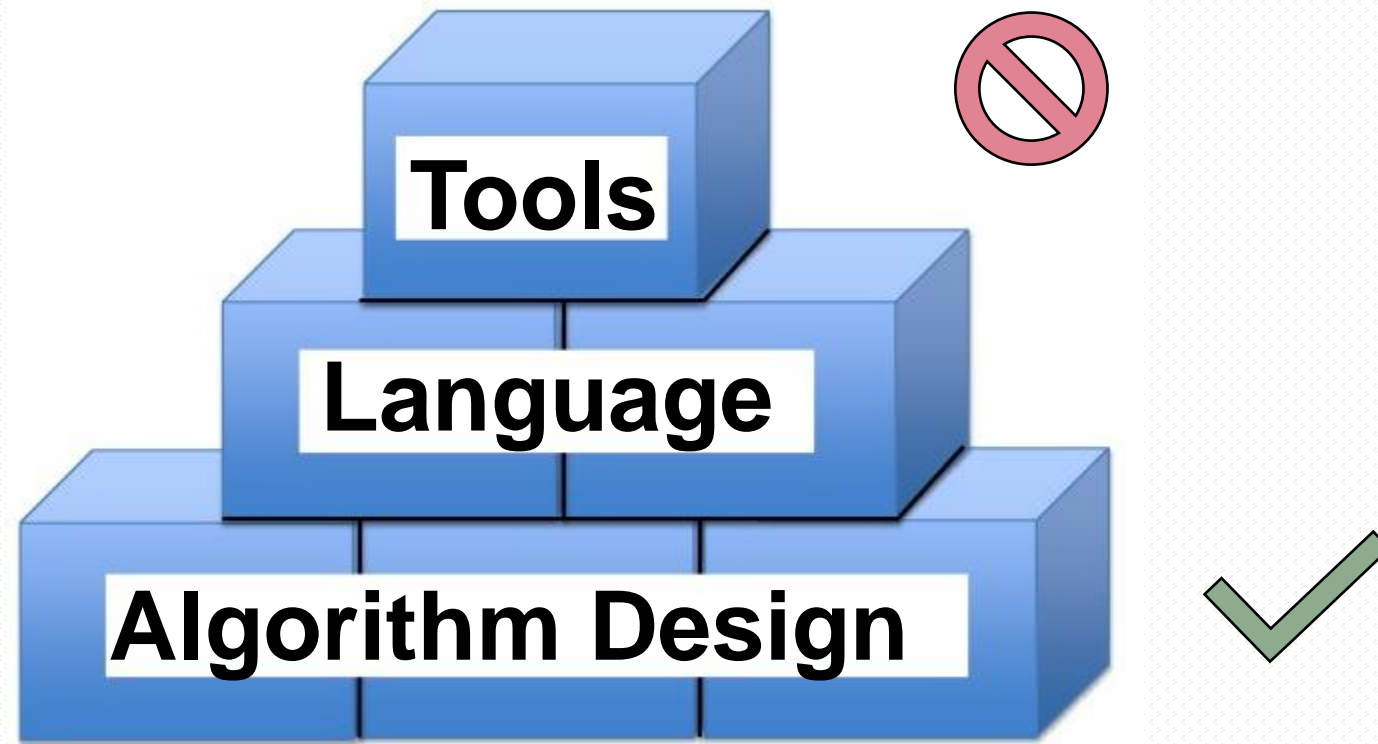
Selenium,	JUnit,	Cucumber,	RestAssured
TestComplete,	NUnit,	SpecFlow,	Soap UI,
UFT,	TestNG,	Serenity,	Chakram,
Telerik,	Jasmine,	Behave,	Jenkins
Robot	Protactor	Gauge	Circle CI



# How to Learn Automation?



# Algorithm Design Aptitude Vs Tools



# Every Human Designs Algorithms



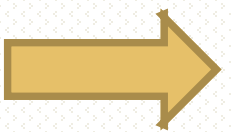
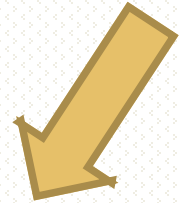
# Every Human Designs Algorithms



# Programmers Create Solutions







```
<?php
$dir = new RecursiveItera
$image = new RegexItera
foreach ($images as $ima
    $files[] = $image->g
}
print_r($files);
```



Create  
Solutions



# Algorithm Design Process




Solve  
problem  
Manually

Write steps  
in detail

Write  
pseudo  
code

Script from  
Pseudo  
code

# Algorithm Design Process



Solve  
problem  
Manually

Write steps  
in detail

Write  
pseudo  
code

Script from  
Pseudo  
code



**Find Age –**







**Current Date - DOB**





# Algorithm Design Process



Solve  
problem  
Manually

Write steps  
in detail





Write  
pseudo  
code

Script from  
Pseudo  
code

**Find Age –**

- 1. Current Date**
- 2. Date of Birth**
- 3. Current - DOB**

# Algorithm Design Process



Solve  
problem  
Manually

Write steps  
in detail

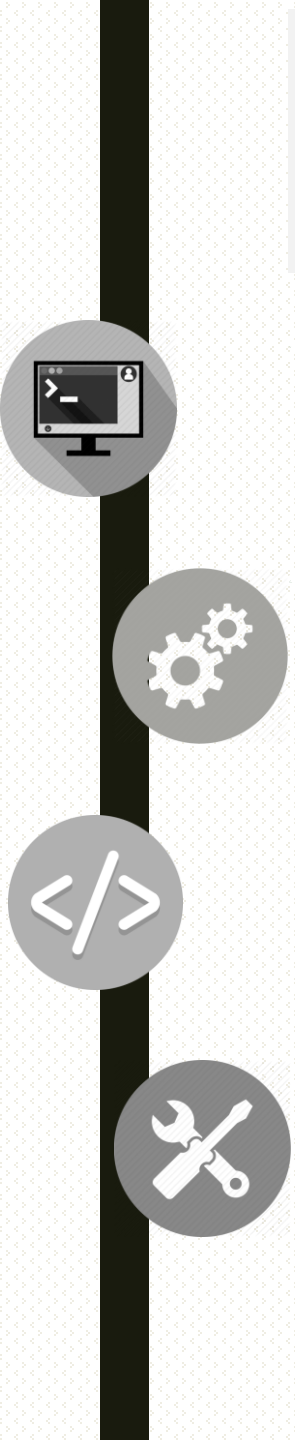
Write  
pseudo  
code

Script from  
Pseudo  
code

## Find Age –

1. **Current Date**
2. **Convert to days**
3. **Date of Birth**
4. **Convert to days**
5. **Current – DOB**
6. **Convert to Years**

# Algorithm Design Process



Solve  
problem  
Manually

Write steps  
in detail

Write  
pseudo  
code

Script from  
Pseudo  
code

## Find Age –

1. **Date** current;
2. **Int** Current = current;
3. **Date** doB;
4. **Int** DoB = doB;
5. **Int** age = Current - DoB
6. **Date** Age = age;

# Algorithm Design Process

Solve problem Manually

**Find Age –**  
**Current Date -**  
**DOB**

Write steps in detail

**Find Age –**

1. **Current Date**
2. **Date of Birth**
3. **Current - DOB**

Write pseudo code

**Find Age –**

1. **Current Date**
2. **Convert to days**
3. **Date of Birth**
4. **Convert to days**
5. **Current – DOB**
6. **Convert to Years**

Script from Pseudo code

**Find Age –**

1. **Date** current;
2. **Int** Current = current;
3. **Date** doB;
4. **Int** DoB = doB;
5. **Int** age = Current - DoB
6. **Date** Age = age;



# Start Easy, Keep Pushing



docker

aws



TestComplete



Jenkins



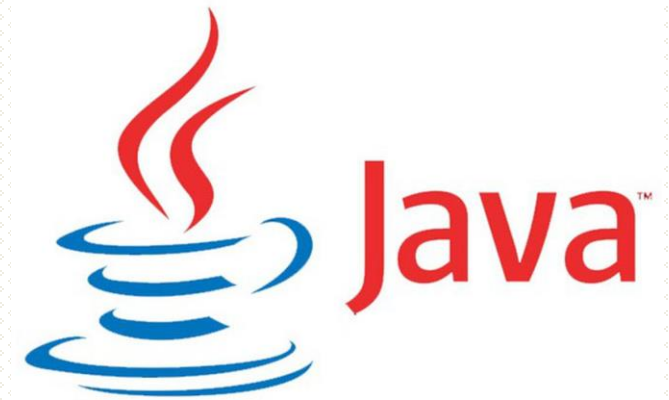
specflow  
Cucumber for .NET

Ju

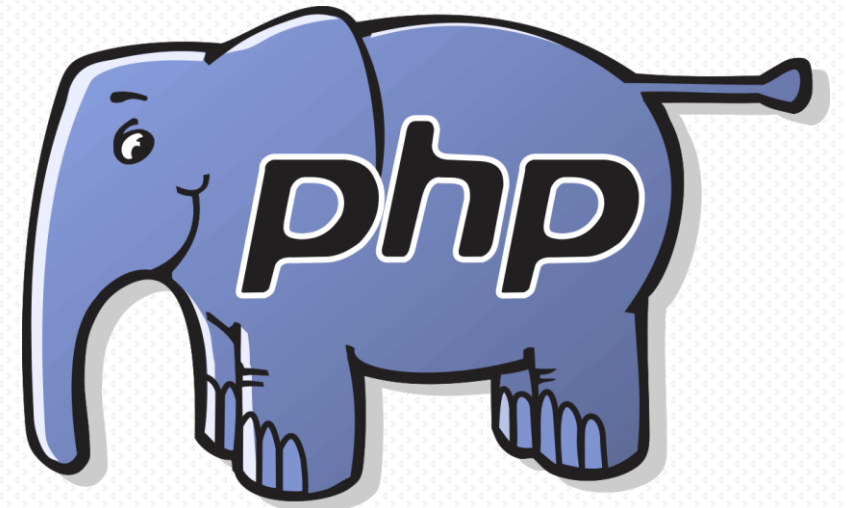
cucumber



# Start Easy, Keep Pushing



**JavaScript**



LESSON  
LEARNED

---

**Develop Aptitude for  
Algorithm Design**



# ASIM'S STRESSFUL DAYS STORY





# Perception of Coder



# Reality of Trouble Shooting



# After Solving the Problem



# How to Deal with Debugging?



# Develop the Attitude



# Attitude is Like Any Muscle



# Enjoy Solving Problems



**LESSON  
LEARNED**

---

**Develop an Attitude to  
Enjoy Solving Problems**





Recap



**Testers must be Technical**

**Test across the Technology Stack**

**Develop aptitude for Algorithm Design**

**Develop an attitude to Enjoy Solving Problems**



**Asim is a**  
**Technical Tester**



Be Like Asim !

# Thank you

PS  
QC'  
18



[linkedin.com/in/alikhalid](https://www.linkedin.com/in/alikhalid)



[Quality-Spectrum.com](https://www.Quality-Spectrum.com)

