



NINJA ★ PROGRAMMER CODING AT THE NIGHT



Agenda

- Introduction**
- Product Backlog**
- Scrum meeting/XP values**
- UML diagrams**
- Architecture diagram**
- Task boards**
- Cumulative Flow Diagram**
- Burndown chart**
- Demo**
- Team retrospective**

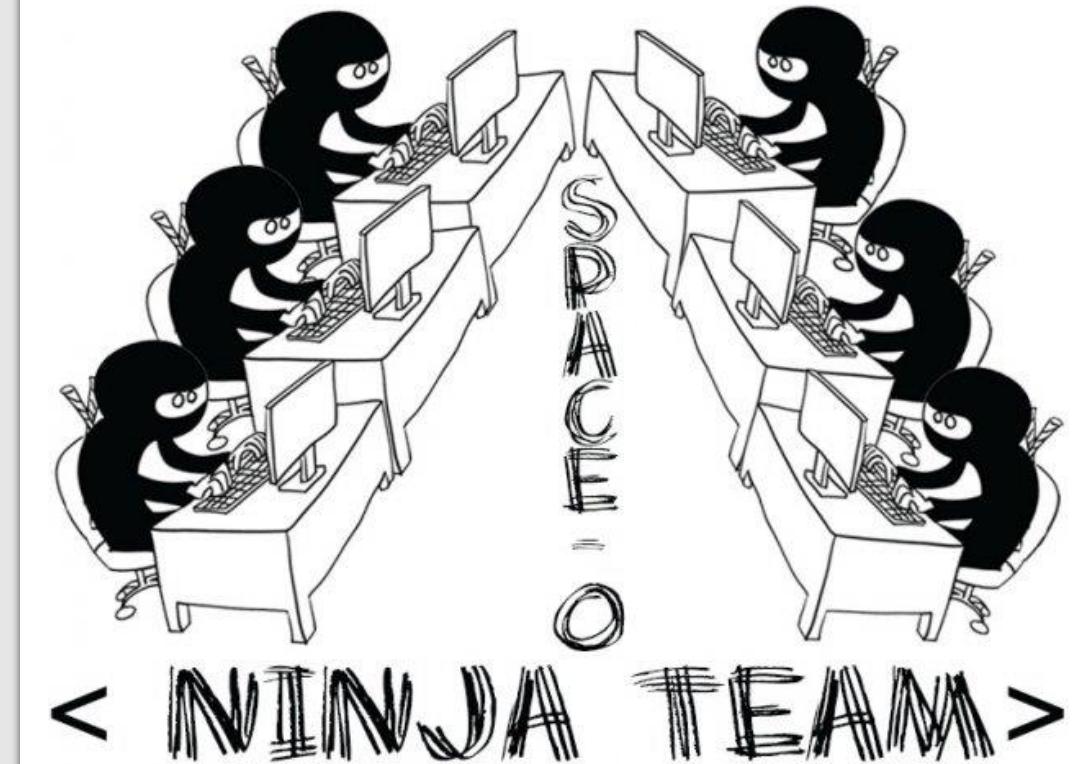
TEAM 1

- Niral Koradia
- Prathmesh Parmar
- Aashish Subramanian
- Neville Mascarenhas
- Abhijitsinh Chavda



Scrum team

- Scrum master:
 - Aashish Subramanian
- Product Owner:
 - Neville Mascarenhas
- Development team:
 - Niral Koradia
 - Prathmesh Parmar
 - Abhijitsinh Chavda



Product Backlog Refinement



Backlog items

Backlog Item	Task	Task Owner
Enable all users to place orders in shopping cart	Design User Interface - Sign Up and Login	Prathmesh
	Implement User Interface Back-End Code	Prathmesh
	Design Product Catalogue	Abhijit
	Implement Product Catalogue Back End Code	Abhijit
	Design Payments and Orders	Aashish
	Implement Orders Back-End Code	Aashish
	Implement Payments Back-End Code	Aashish
	Design Shopping Cart	Niral
	Implement Shopping Cart Back-End Code	Niral
	Design Logging	Neville
Creating UML Class Design	Implement Logging Back-End Code	Neville
	User Sessions	Prathmesh
	Product Catalog	Abhijit
	Shopping Cart	Niral
	Payments and Orders	Aashish
	Logging	Neville

Sprint #1

Weekly Scrum #1 – 10/22/1017

Team Member Name: Aashish Subramanian

What I did since the last weekly scrum:

- Designed the architecture for the payments and orders sub-modules
- Designed the schema structure for the data
- Designed the UI skeletal

What I plan to do this week:

- Implement the MongoDB replication mechanism and create the required storage structure for data
- Implement the backend API services in Golang and ensure connectivity with the MongoDB replica

What Blockers I have:

- None

Sprint #2

Weekly Scrum #2 – 10/29/1017

Team Member Name: Abhijitsinh Chavda

What I did since the last weekly scrum:

- Created MongoDB replica set with database and collections for product catalogue
- Created Nodejs APIs to retrieve data from MongoDB
- Performed unit test to ensure connectivity of API to MongoDB on partition failure

What I plan to do this week:

- Implement the front end for Product Catalogue using Nodejs and express

What Blockers I have:

- Configuration issue in MongoDB replica-set due to which Primary failed to be elected

Team Scrum Meeting

Sprint #3

Weekly Scrum #3 – 11/05/1017

Team Member Name: Niral Koradia

What I did since the last weekly scrum:

- Implemented front end for shopping cart module
- Ensured connectivity from frontend to backend node services for shopping cart

What I plan to do this week:

- Implement group cart feature in the shopping cart

What Blockers I have:

Selecting the sharing strategy for groups

Sprint #4

Weekly Scrum #4 – 11/12/1017

Team Member Name: Neville Mascarenhas

What I did since the last weekly scrum:

- Performed deployment of logging sub-module to AWS
- Performed unit test to ensure connectivity from frontend
- Displayed logs in the Admin dashboard

What I plan to do this week:

- Perform integration testing

What Blockers I have:

- Port related issues on AWS which were eventually resolved

Team Scrum Meeting

Sprint #4

Weekly Scrum #4 – 11/12/1017

Team Member Name: Prathmesh Parmar

What I did since the last weekly scrum:

- Performed sub-module integrations
- Performed unit and integration tests on all modules

What I plan to do this week:

- Complete the UML related activities for the project

What Blockers I have:

- None

Team Scrum Meeting

Simplicity

Communication

Eliminate Waste

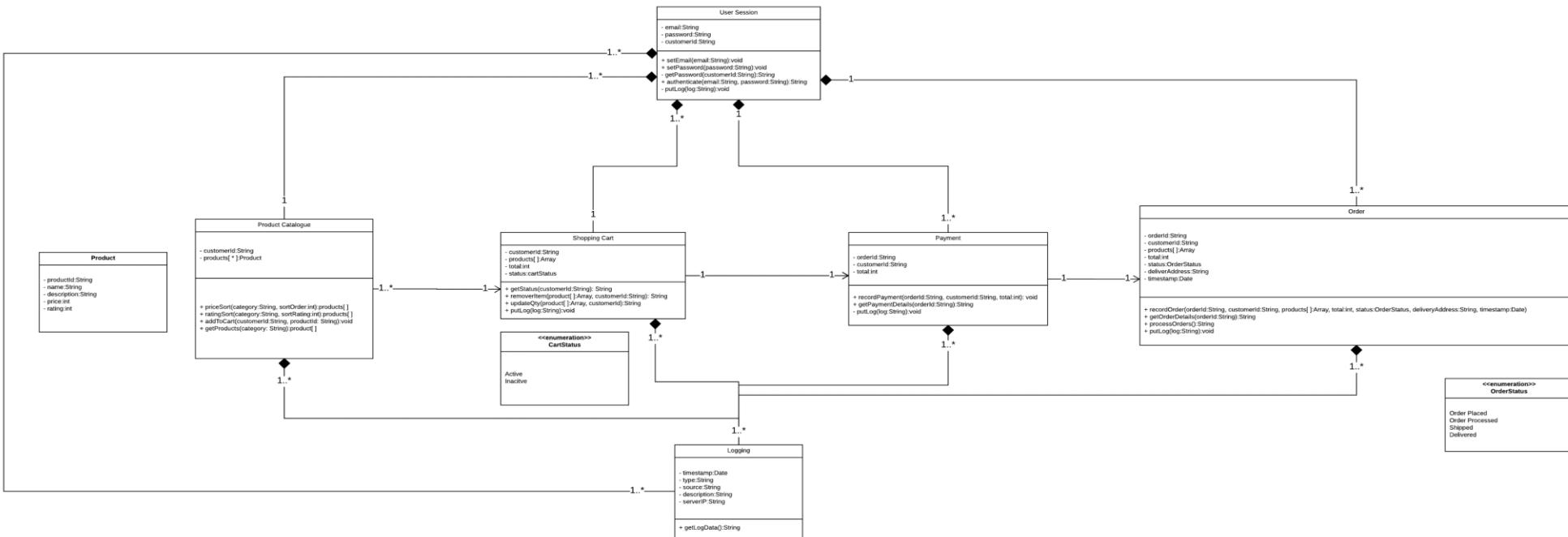
See the whole

Feedback

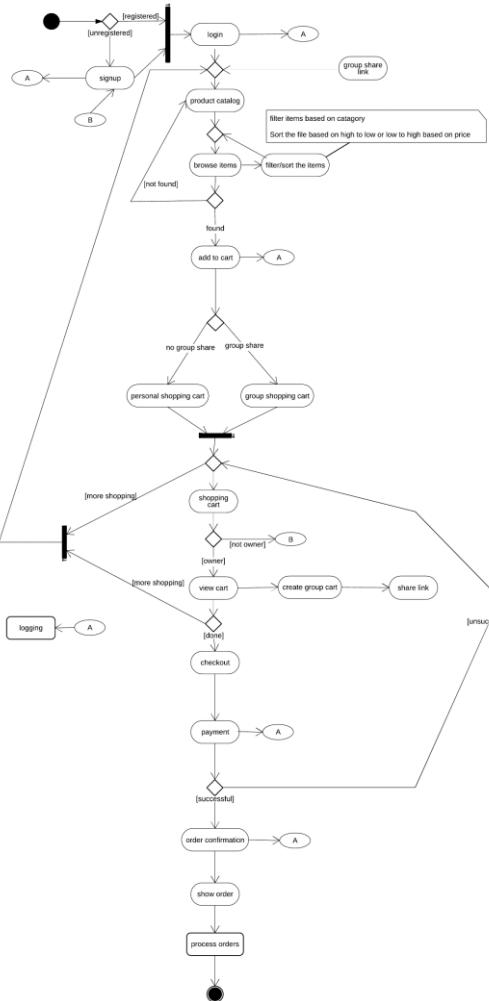
Core Values

UML Diagrams

Class Diagram



Activity Diagram



User Story

As a user, I would like to have a shopping cart application where I or my group users can search for all the items in the product catalogue, see its description, quantities, price, etc. in details, can add or remove from the cart along with the option to share the same cart with friends and have facilities to authenticate the payment and process the orders.

#User Stories - Overview

- 1) Authentication: User can signup/login/logout.
- 2) Items for sale: User can view lists of items for sale and search by name, filter by price.
- 3) Shopping Cart: User/group of users can add items to shopping cart and the app remembers it next time you login. Users can view all the items in their shopping cart. Users can delete items in the shopping cart.
- 4) Checkout: User(s) can fill in form and submit billing info. After submitting billing info, items in the shopping cart should be processed.

User Story

Authentication

Registration:

- 1) Create a sign in/signup page.
- 2) Add a URL/controller/template /signup.
- 3) /signup has a form, username, email, and password.
- 4) "Submit" button posts to /register.
- 5) /register creates a new user.

Authenticate:

- 1) Create a new page that is only for logged in users.
- 2) If the user is logged in, show the page.
- 3) If not, redirect the user to the login page

User Story

Items

1) Create a new Item Model with the following fields:

Name, Description, Price

2) Create several in the admin or shell

3) Create new routes and templates to show a listing of the items

Filter

Allow the user to filter items by price.

Shopping cart/order

An order belongs to a user or group of users, and has multiple items. A user can have many orders.

Allow user/users of same group can add/delete items from the cart

Shows the total price of all items

Allows them to purchase items, purchasing takes the user to payment form at /payments

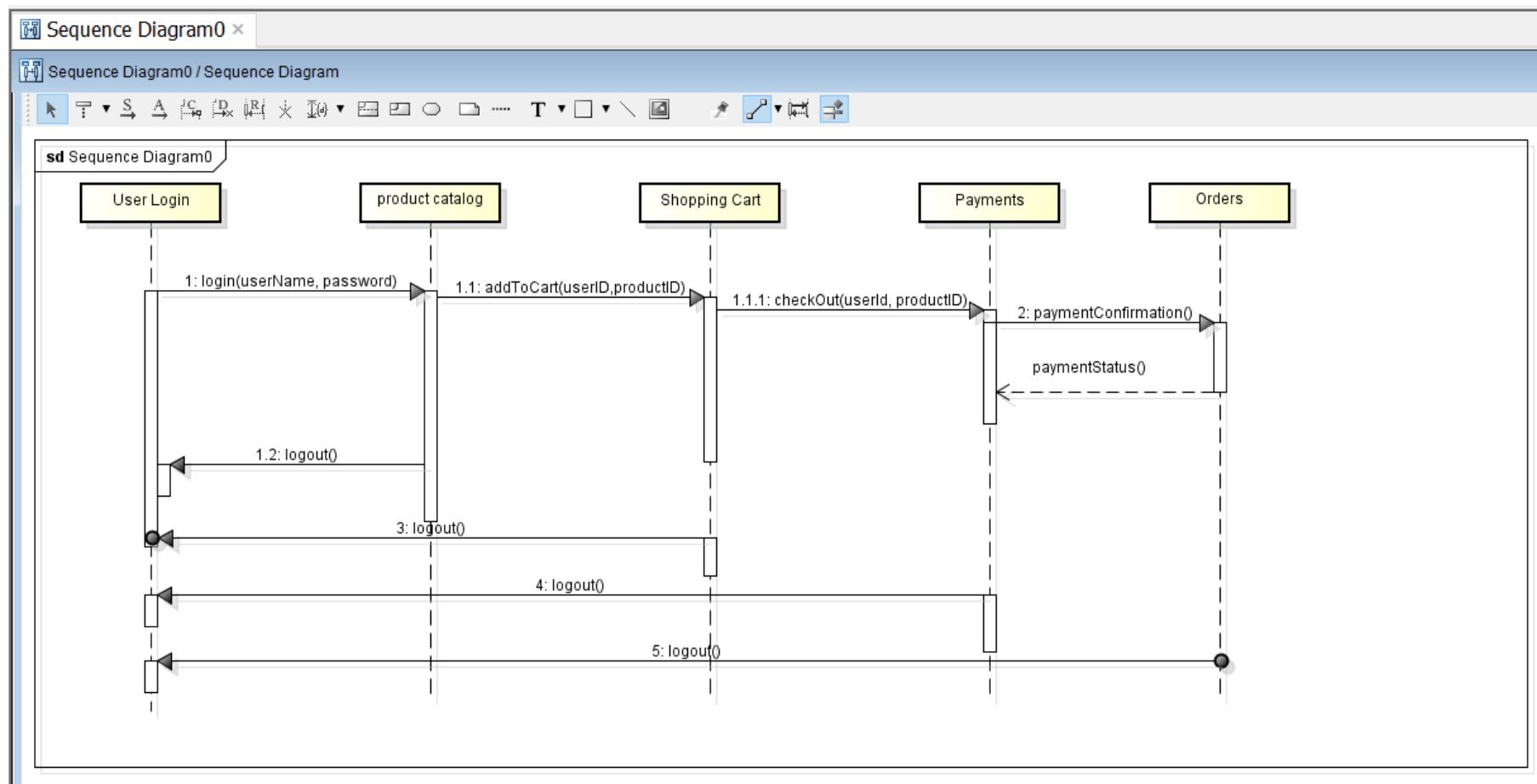
User Story

Payment form

Create a form that allows the user to enter billing info

On submit, the order should change to processed and inventory should be updated properly.

Sequence Diagram



UI-WireFrames - Login

Login

70 x 70 Label

Home About Orders 31 X 30 Contact Us Hi Guest!

70 x 70 Label

31 X 30 Contact Us

Hi Guest!

Login

Username

Email/Phone

Password

[Forgot Password](#)

Login

Signup

Label

Home About Help Contact Us

UI-WireFrames - SignUp

Signup

70 x 70 Label

Home About Orders **\$1 X /30s** Contact Us Hi Guest!

Signup

First name

Last name

Email

Phone number

Register

Label

Home About Help Contact Us

UI-WireFrames – Product Catalogue

Product Catalogue

70 x 70 Label

Home About Orders 31 x 30 Contact Us Hi Username!

Category Name

Category1

Category2

Category3

Category4

Category5

Group Cart

Image	Product name	Price	Rating	Add to Cart
82 x 128	Product name _____	—	—	Add to Cart
82 x 128	Product name _____	—	—	Add to Cart
82 x 128	Product name _____	—	—	Add to Cart
82 x 128	Product name _____	—	—	Add to Cart
82 x 128	Product name _____	—	—	Add to Cart
82 x 128	Product name _____	—	—	Add to Cart
82 x 128	Product name _____	—	—	Add to Cart
82 x 128	Product name _____	—	—	Add to Cart
82 x 128	Product name _____	—	—	Add to Cart

Label

Home About Help Contact Us

UI-WireFrames - ShoppingCart

Shopping Cart

70 x 70 Label

Home About Orders \$1 X /30 Contact Us Hi Username!

Login validity message

Product name
100 x 100
Qty MenuItem
MenuItem
MenuItem

Price —

Product name
100 x 100
Qty MenuItem
MenuItem
MenuItem

Price —

Product name
100 x 100
Qty MenuItem
MenuItem
MenuItem

Price —

Total —

Continue Shopping Checkout

Label

Home About Help Contact Us

UI-WireFrames - CheckOut

Checkout

70 x 70 Label

Home About Orders 31 x 30 Contact Us Hi Username!

Address Details

Name

Address

City Pin

Name on Card

Card Number

mm/yy CVV

Continue Shopping Pay Now

Label

Home About Help Contact Us

UI-WireFrames – Admin Console

Admin Console

The wireframe illustrates the layout of the Admin Console. At the top, there is a header section with a logo placeholder (70x70 pixels) labeled "Label". To the right of the logo are navigation links: Home, About, Orders, Contact Us, and Hi Admin. Below the header is a large central area labeled "Logs" containing a rectangular text input field with five horizontal lines. At the bottom, there is a footer section with a "Label" placeholder and a row of navigation links: Home, About, Help, and Contact Us.

70 x 70 Label

Home About Orders Contact Us Hi Admin

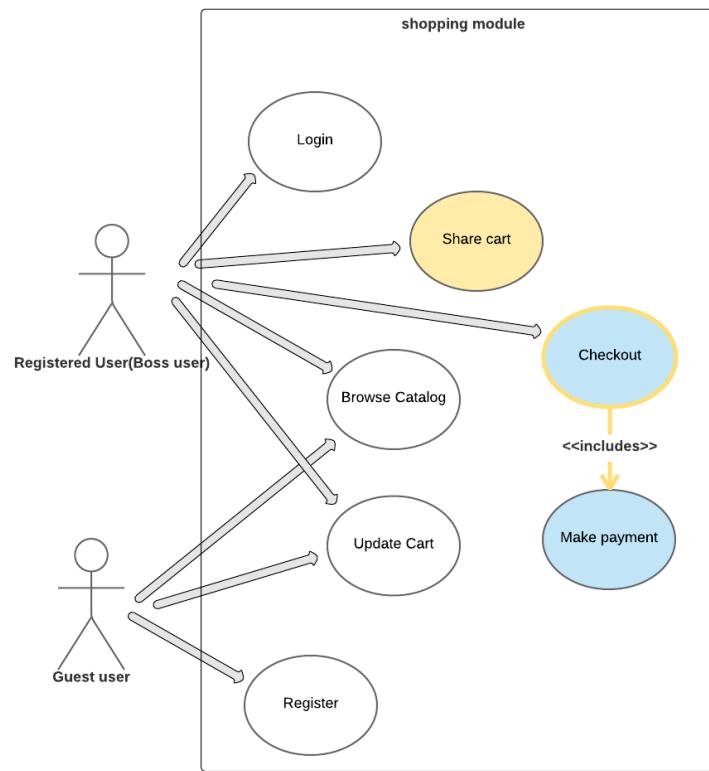
Logs

Label

Home About Help Contact Us

Use-Case Diagram

GROCERY ORDERING SYSTEM



Use-Case Specification

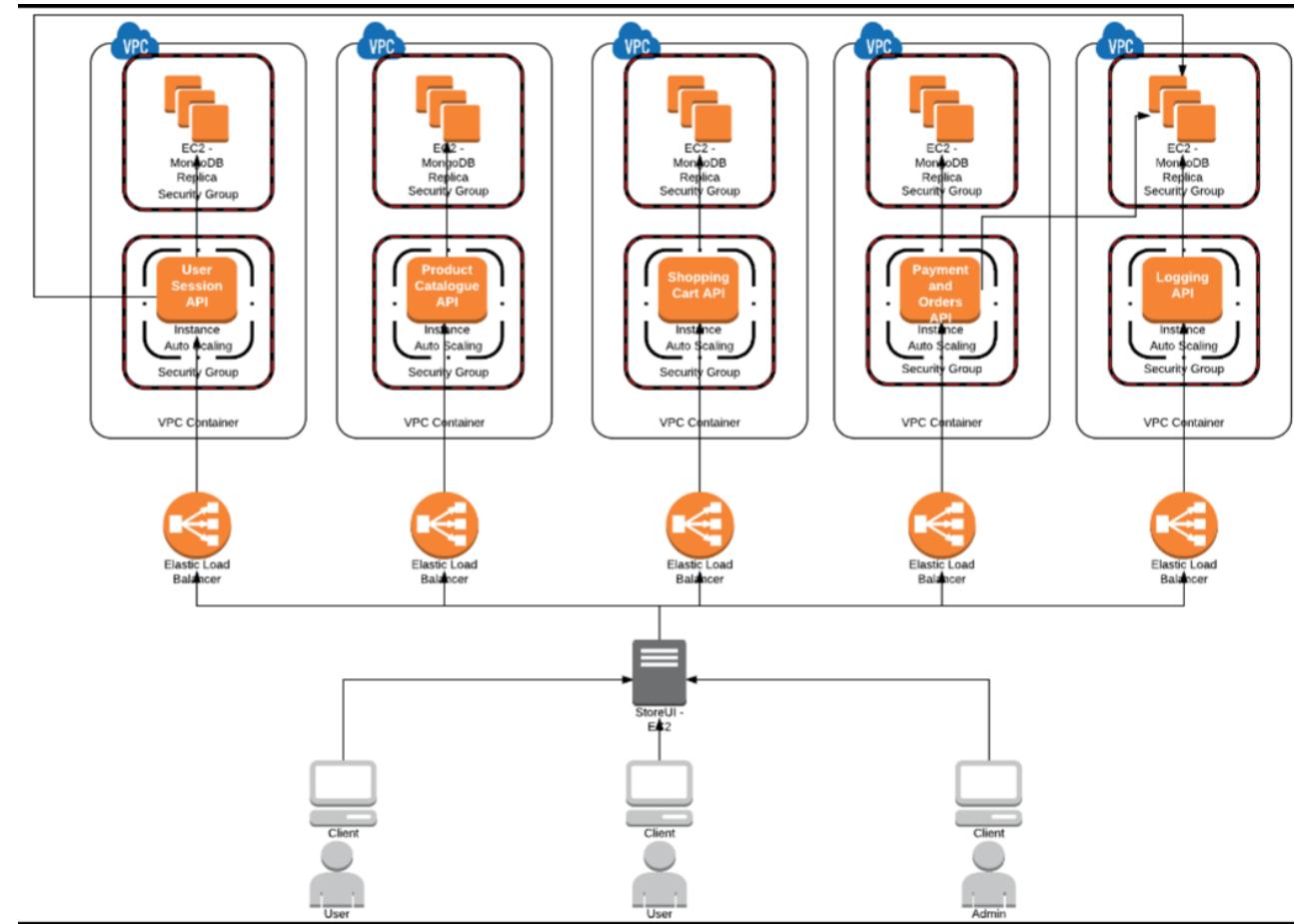
Ninja

The following are the core use cases for the Grocery ordering application.

Use Cases

Sr.No	Use Case	Module	Description
1.	New User Registration	User Management	<ul style="list-style-type: none">User is directed to the sign up pageUser enters required detailsUser is directed to the shopping cart on a new session after the details are recorded
2.	Existing User login/logout	User Management	<ul style="list-style-type: none">User is directed to the login pageUser enters their ID and passwordId and password are validatedSuccessful login takes the user to the product catalogFailure of validation takes the user back to the login page
3.	Search Product Catalog	Product Catalog	<ul style="list-style-type: none">Valid user can view the product catalogUser can switch between categories available in a left pane menu
4.	Sort Products	Product Catalog	<ul style="list-style-type: none">Valid user can select a sort option from the left pane menuSorting can be done based on price and ratingsUser can specify the sorting order to be either ascending or descending
5.	Add product to cart	Shopping cart	<ul style="list-style-type: none">The user can click the <i>Add to cart</i> button which will add the product to the cart.User can also click on the <i>Remove</i> button if they wish to remove the item from their shopping cart.
6.	Checkout the product(Buy)	Shopping cart	<ul style="list-style-type: none">If the user wishes to complete the shopping and proceed to the payment he can click the <i>Checkout</i> button to proceed to the payment.
7.	Make Payment	Payments	<ul style="list-style-type: none">The <i>Payment</i> button will allow the user to complete the payment as per their choice.
8.	Complete order process	Payments	<ul style="list-style-type: none">Once the payment is successfully processed the user will receive an order id.
9.	Logging data	Logging	<ul style="list-style-type: none">The data events from the modules will be logged into the data store.
10.	Data analytics	Logging	<ul style="list-style-type: none">The dashboard will display various details for the grocery application.

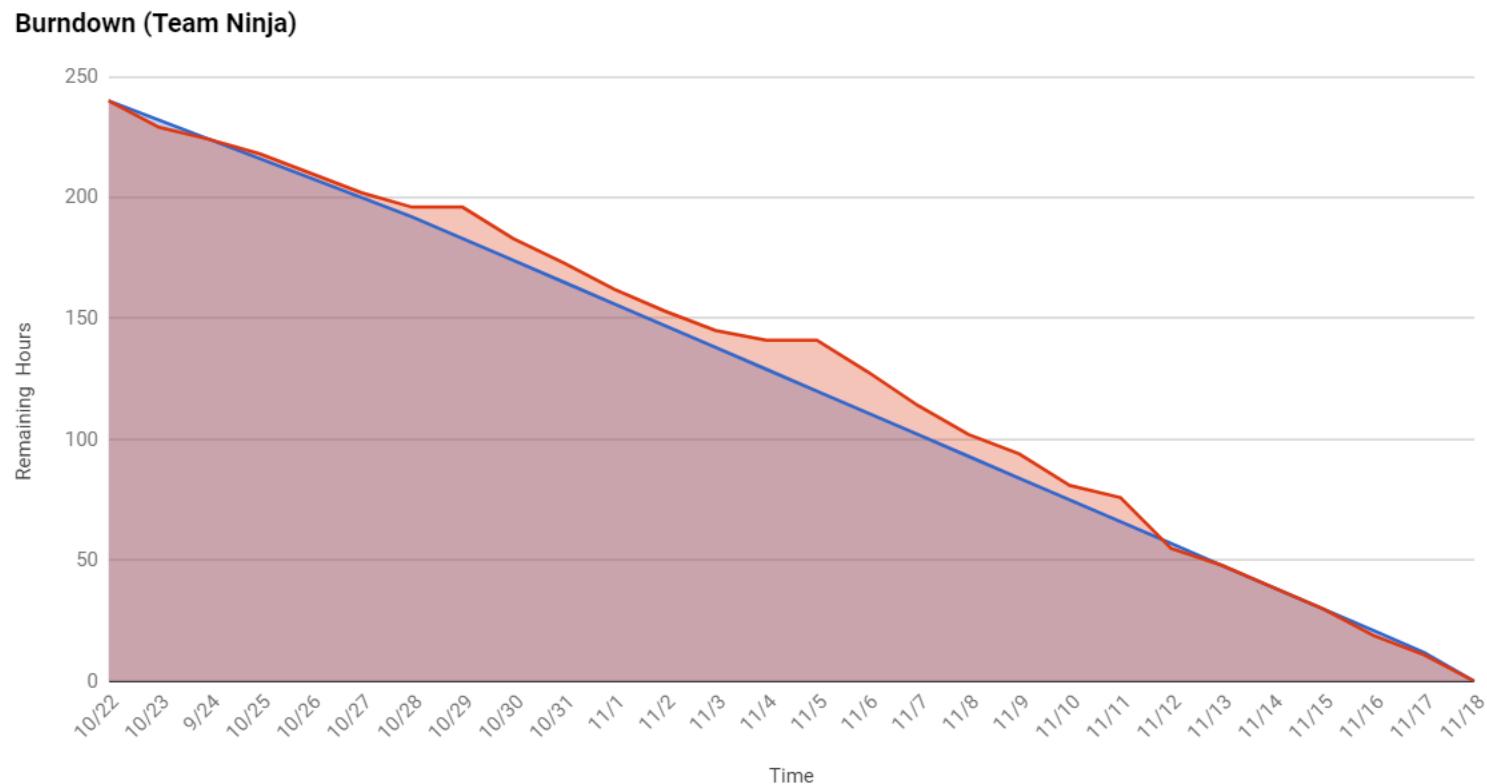
Architecture Diagram



Task Board

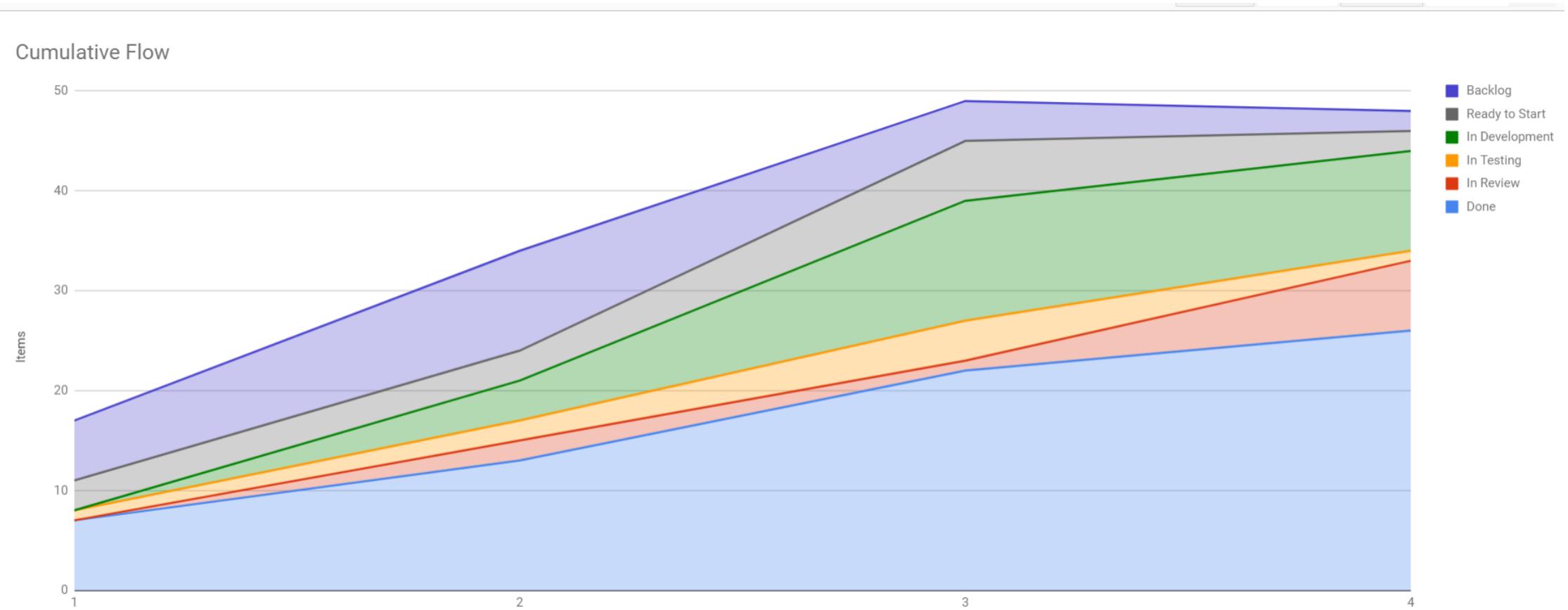
<https://github.com/nguyensjsu/cmpe281-ninja/projects/1>

Burndown chart



<https://docs.google.com/spreadsheets/d/1a6EvbuCKLuY8zbLjcvXbUJzA-U4jcad7dhJigOqlIZ4/edit#gid=0>

Cumulative Flow Diagram



<https://docs.google.com/spreadsheets/u/1/d/1KnVS7Ho-DIToUMpmldLaP6jSARa9GZJmGSTukzRc008Y/edit?usp=sharing>

Resource Groups

Create Security Group Actions

Filter by tags and attributes or search by keyword

1 to 4 of 4

Name	Group ID	Group Name	VPC ID	Description
sg-2bcc7e4d		default	vpc-7a672c1e	default VPC security group
sg-6a78c70c		cmpe281-dmz	vpc-7a672c1e	cmpe281-dmz created 2017-09-04T15:52:...
sg-918584f7		mongo-cluster	vpc-ba9b64de	launch-wizard-1 created 2017-11-29T15:2...
Type	Protocol	Port Range	Source	Description
HTTP	TCP	80	0.0.0.0/0	
HTTP	TCP	80	::/0	
Custom TCP Rule	TCP	27019	0.0.0.0/0	
Custom TCP Rule	TCP	27018	0.0.0.0/0	
Custom TCP Rule	TCP	7001	0.0.0.0/0	
SSH	TCP	22	0.0.0.0/0	
Custom TCP Rule	TCP	27017	0.0.0.0/0	
Custom TCP Rule	TCP	27017	::/0	
Custom TCP Rule	TCP	3000	0.0.0.0/0	

Launch Instance Connect Actions

Filter by tags and attributes or search by keyword

1 to 6 of 6

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
mongo-1	i-02ab40cb2fb76f41a	t2.micro	us-west-1a	running	2/2 checks ...	None	ec2-54-183-129-56.us-west-1.co...
backend	i-02f4912d490aa2b60	t2.micro	us-west-1a	running	2/2 checks ...	None	ec2-54-153-39-16.us-west-1.co...
scaled-api-backend	i-09a69ebf03ffcb731	t2.micro	us-west-1a	running	2/2 checks ...	None	ec2-54-153-51-203.us-west-1.co...
mongo-0	i-0be7bbc0ac2553cab	t2.micro	us-west-1a	running	2/2 checks ...	None	ec2-54-241-134-246.us-west-1.c...
NinjaGrocery-FrontEnd	i-0e1762cce20964890c	t2.micro	us-west-1a	running	2/2 checks ...	None	ec2-52-53-172-74.us-west-1.co...
mongo-2	i-0f69d88e0f7b272ff	t2.micro	us-west-1a	running	2/2 checks ...	None	ec2-54-183-244-236.us-west-1.c...

User Session – Security Group & EC2

The screenshot shows the AWS Load Balancer console. At the top, there's a search bar and navigation links for Prathmesh Parmar, N. California, and Support. Below that, a table lists existing load balancers. A new load balancer, 'cmpe281-loadbalancer', is being created, indicated by the blue selection box. The 'Basic Configuration' section displays the following details:

Name	DNS name	State	VPC ID	Availability Zones	Type
aws-php-elb-classic	aws-php-elb-classic-194026...	Active	vpc-7a672c1e	us-west-1a, us-west-1b	classic
cmpe281-loadbalancer	cmpe281-loadbalancer-1689084271.us-west-1.elb.amazonaws.com (A Record)	Creating	vpc-ba9b84de	us-west-1a, us-west-1b	classic

Basic Configuration

Name: cmpe281-loadbalancer	Creation time: December 1, 2017 at 12:12:57 AM UTC-8
* DNS name: cmpe281-loadbalancer-1689084271.us-west-1.elb.amazonaws.com (A Record)	Hosted zone: Z368ELLRRE2KJ0
Type: Classic (Migrate Now)	Status: 2 of 2 instances in service
Scheme: internet-facing	VPC: vpc-ba9b84de
Availability Zones: subnet-b49a53ef - us-west-1a, subnet-fd06bf9a - us-west-1b	

Port Configuration

English (US) © 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

The screenshot shows the AWS Auto Scaling console. At the top, there's a search bar and navigation links for Prathmesh Parmar, N. California, and Support. Below that, a table lists existing Auto Scaling groups. An Auto Scaling group, 'cmpe281-project', is being edited, indicated by the blue selection box. The 'Launch Configuration' section displays the following details:

Name	Launch Configuration /	Instances	Desired	Min	Max	Availability Zones	Default Cooldown	Health Check Grace Period
cmpe281-project	cmpe281-project	1	1	1	2	us-west-1a	300	299
aws-php-autoscale	aws-php-autoscale	0	0	0	0	us-west-1a, us-west-1b	300	300

Launch Configuration cmpe281-project

Launch Template

Launch Template Version

Load Balancers	cmpe281-loadbalancer
Target Groups	
Desired	1
Min	1
Max	2

Health Check Type EC2

Health Check Grace Period 299

Termination Policies Default

Creation Time Thu Nov 30 23:54:49 GMT-800 2017

Availability Zone(s) us-west-1a
Subnet(s) subnet-b49a53ef
Default Cooldown 300
Placement Group

Suspended Processes

Enabled Metrics

Instance Protection

User Session-Load Balancer & Auto Scaling Group

The screenshot shows the AWS Auto Scaling Groups page. At the top, there's a search bar and a filter section. Below it, a table lists one Auto Scaling Group: 'product_catalog_server'. The group has 1 instance, desired count 1, and minimum/maximum of 1. It's configured for two availability zones: us-east-2a and us-east-2b, with a default cooldown of 300 seconds and a health check grace period of 300 seconds. The launch configuration is 'product_catalog_server_launch'. The details tab is selected, showing the launch configuration, target groups (with desired count 1), and placement group. The load balancer is 'product-catalog-server-load'. The creation time was Thu Nov 30 00:49:26 GMT-000 2017.

The screenshot shows the AWS Load Balancers page. At the top, there's a search bar and a filter section. Below it, a table lists one load balancer: 'product-catalog-server-load'. It has a DNS name of product-catalog-server-load-... and is associated with the VPC ID vpc-f3486ed9, located in the us-east-2a and us-east-2b availability zones. The type is classic, and it was created on November 30, 2017, at 12:56:00. The connection draining is enabled with a timeout of 300 seconds. The load balancer is currently managing two instances: 'product_server_auto' and 'prodserver', both in the us-east-2a availability zone and in service. The instance protection is set to Enabled Metrics. The availability zones table shows two subnets: subnet-3714225e in us-east-2a and subnet-ac1d2d7 in us-east-2b.

Product Catalog-Load Balancer & Auto Scaling Group

AWS Services Resource Groups

EC2 Dashboard

Create Security Group Actions

Filter by tags and attributes or search by keyword

Name	Group ID	Group Name	VPC ID	Description
sg-02bf263	cmpe281-dmz	vpc-f54865ea	launch-wizard-1 created 2017-09-04T14:31:09.921-07:00	
Custom UDP Rule	UDP	6783	0.0.0.0/0	
Custom UDP Rule	UDP	6783	0.0.0.0/0	
HTTP	TCP	80	0.0.0.0/0	
HTTP	TCP	80	0.0.0.0/0	
Custom TCP Rule	TCP	27019	0.0.0.0/0	
Custom TCP Rule	TCP	27019	0.0.0.0/0	
Custom TCP Rule	TCP	27018	0.0.0.0/0	
Custom TCP Rule	TCP	27018	0.0.0.0/0	
Custom TCP Rule	TCP	90	0.0.0.0/0	
Custom TCP Rule	TCP	90	0.0.0.0/0	
SSH	TCP	22	0.0.0.0/0	
SSH	TCP	22	0.0.0.0/0	
Custom TCP Rule	TCP	6379	0.0.0.0/0	
Custom TCP Rule	TCP	6379	0.0.0.0/0	
Custom TCP Rule	TCP	6783	0.0.0.0/0	
Custom TCP Rule	TCP	6783	0.0.0.0/0	
Custom TCP Rule	TCP	27017	0.0.0.0/0	
Custom TCP Rule	TCP	27017	0.0.0.0/0	
Custom TCP Rule	TCP	2375	0.0.0.0/0	
Custom TCP Rule	TCP	2375	0.0.0.0/0	
Custom TCP Rule	TCP	3000	0.0.0.0/0	
Custom TCP Rule	TCP	3000	0.0.0.0/0	
HTTPS	TCP	443	0.0.0.0/0	
HTTPS	TCP	443	0.0.0.0/0	

Feedback English (US)

© 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

AWS Services Resource Groups

EC2 Dashboard

Launch Instance Actions

Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP	IPv6 IPs	Key Name	Monitoring	Launch Ti
aws-php	i-039e6049578118837	t2.micro	us-east-2a	stopped	0/2 checks ...	None	-	-	-	ec2php	disabled	November
product_serv...	i-03e14368f014265d5	t2.micro	us-east-2a	running	2/2 checks ...	None	ec2-18-220-224-28.us...	18.220.224.28	-	ec2php	disabled	November
prodmongo1	i-059126d13ab028	t2.micro	us-east-2a	running	2/2 checks ...	None	ec2-18-217-172-152.us...	18.217.172.152	-	ec2php	disabled	November
prodserver	i-09911c9f007270ae	t2.micro	us-east-2a	running	2/2 checks ...	None	ec2-18-217-169-200.us...	18.217.169.200	-	ec2php	disabled	November
prodmongo2	i-0c96c1070940d532	t2.micro	us-east-2a	running	2/2 checks ...	None	ec2-18-216-213-231.us...	18.216.213.231	-	ec2php	disabled	November
prodmongo3	i-0d951cd756848352	t2.micro	us-east-2a	running	2/2 checks ...	None	ec2-18-217-114-5.us...	18.217.114.5	-	ec2php	disabled	November

Select an instance above

Product Catalog – EC2 & Scaling Group

EC2 Management Console - Google Chrome

Secure | https://us-west-1.console.aws.amazon.com/ec2/v2/home?region=us-west-1#instances:sort=instanceid

Apps Home Bookmarks Node.js Tutorial mean_mytasklist Colang tutorial cmpe281/globa Jobs DeepMask Inst...

Services Resource Groups

Neal Korada N. California Support

EC2 Dashboard Events Tags Reports Limits

INSTANCES

Instances Launch Templates Spot Requests Reserved Instances Dedicated Hosts

IMAGES

AMIs Bundle Tasks

ELASTIC BLOCK STORE

Volumes Snapshots

NETWORK & SECURITY

Security Groups Elastic IPs Placement Groups Key Pairs Network Interfaces

LOAD BALANCING

Load Balancers Target Groups

AUTO SCALING

Feedback English (US)

© 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

EC2 Management

Launch Instance Connect Actions

Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
shopping-cart-node	i-034d21c5d7358e25	t2.micro	us-west-1c	running	2/2 checks ...	None	ec2-13-56-155-199.us-west-1.compute.amazonaws.com
mongodb-27019	i-039d423b3cd3736d0	t2.micro	us-west-1c	running	2/2 checks ...	None	ec2-13-56-227-70.us-west-1.compute.amazonaws.com
mongodb-27018	i-05714fbf9303050ca	t2.micro	us-west-1c	running	2/2 checks ...	None	ec2-13-56-159-30.us-west-1.compute.amazonaws.com
shopping-cart-node-2	i-07ce4ce752735ba96	t2.micro	us-west-1c	running	2/2 checks ...	None	ec2-13-54-193-113-183.us-west-1.compute.amazonaws.com
mongodb-27017	i-0ab37429ab9f6336	t2.micro	us-west-1c	running	2/2 checks ...	None	ec2-13-56-194-186.us-west-1.compute.amazonaws.com

Select an instance above

EC2 Management Console - Google Chrome

Secure | https://us-west-1.console.aws.amazon.com/ec2/v2/home?region=us-west-1#securityGroups:sort=groupid

Apps Home Bookmarks Node.js Tutorial mean_mytasklist Colang tutorial cmpe281/globa Jobs DeepMask Inst...

Services Resource Groups

Neal Korada N. California Support

EC2 Dashboard Events Tags Reports Limits

SECURITY GROUPS

Create Security Group Actions

Filter by tags and attributes or search by keyword

Name	Group ID	Group Name	VPC ID	Description
sg-7099513	cmpe281-dmz	vpc-e2b1206	vpc-e2b1206	Security Group for cmpe281-dmz
sg-02f5286	default	vpc-e2b1206	vpc-e2b1206	default VPC security group

Security Group: sg-7099513

Edit

Type	Protocol	Port Range	Source	Description
HTTP	TCP	80	0.0.0.0/0	
HTTP	TCP	80	0.0.0.0/0	
Custom TCP Rule	TCP	9000	0.0.0.0/0	
Custom TCP Rule	TCP	9000	0.0.0.0/0	
Custom TCP Rule	TCP	27019	0.0.0.0/0	
Custom TCP Rule	TCP	27019	0.0.0.0/0	
Custom TCP Rule	TCP	27018	0.0.0.0/0	
Custom TCP Rule	TCP	27018	0.0.0.0/0	
Custom TCP Rule	TCP	90	0.0.0.0/0	
Custom TCP Rule	TCP	90	0.0.0.0/0	
SSH	TCP	22	0.0.0.0/0	
SSH	TCP	22	0.0.0.0/0	
Custom TCP Rule	TCP	4000	0.0.0.0/0	
Custom TCP Rule	TCP	4000	0.0.0.0/0	
Custom TCP Rule	TCP	27017	0.0.0.0/0	
Custom TCP Rule	TCP	27017	0.0.0.0/0	
Custom TCP Rule	TCP	3000	0.0.0.0/0	
Custom TCP Rule	TCP	3000	0.0.0.0/0	
HTTPS	TCP	443	0.0.0.0/0	
HTTPS	TCP	443	0.0.0.0/0	

Shopping cart – Security Group & EC2

EC2 Management Console - Google Chrome

EC2 Management C https://us-west-1.console.aws.amazon.com/ec2/v2/home?region=us-west-1#LoadBalancers:

Niraj Koradia N. California Support

Services Resource Groups Actions

Create Load Balancer

Filter: Search 1 to 1 of 1

Name	DNS name	State	VPC ID	Availability Zones	Type
Ninja-SC-load	Ninja-SC-load-108525937.us-west-1.elb.amazonaws.com	Active	vpc-e2efc286	us-west-1b, us-west-1c	classic

Load balancer: Ninja-SC-load

Instances Health Check Listeners Monitoring Tags Migration

Connection Draining: Enabled, 300 seconds (Edit)

Edit Instances

Instance ID	Name	Availability Zone	Status	Actions
i-034d215d7358e255	shopping-cart-node	us-west-1c	InService	Remove from Load Balancer
i-07ce4ce752735ba96	shopping-cart-node-2	us-west-1c	InService	Remove from Load Balancer

Edit Availability Zones

Availability Zone	Subnet ID	Subnet CIDR	Instance Count	Healthy?	Actions
us-west-1b	subnet-994ebc1	10.0.2.0/24	0	No (Availability Zone contains no healthy targets)	Remove from Load Balancer
us-west-1c	subnet-12f63f75	10.0.0.0/24	2	Yes	Remove from Load Balancer

Feedback English (US) © 2008-2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

EC2 Management Console - Google Chrome

EC2 Management C https://us-west-1.console.aws.amazon.com/ec2/autoscaling/home?region=us-west-1#AutoScalingGroupsId=Ninja-Shopping-Cart-AutoScalingview=details

Niraj Koradia N. California Support

Services Resource Groups Actions

Create Auto Scaling group

Filter: Filter Auto Scaling groups... 1 to 2 of 2 Auto Scaling Groups

Name	Launch Configuration	Instances	Desired	Min	Max	Availability Zones	Default Cooldown	Health Check Grace Period
Ninja-Shopping-Cart-Auto-Scaling	Ninja-Shopping-Cart	1	1	1	3	us-west-1b, us-west-1c	300	300
aws-php-autoscale	aws-php-autoscale	0	0	0	0	us-west-1b, us-west-1c	300	300

Auto Scaling Group: Ninja-Shopping-Cart-Auto-Scaling

Details Activity History Scaling Policies Instances Monitoring Notifications Tags Scheduled Actions Lifecycle Hooks Edit

Launch Configuration Ninja-Shopping-Cart

Launch Template Version

Load Balancers Ninja-SC-load

Target Groups

Desired	Min	Max
1	1	3

Health Check Type EC2

Health Check Grace Period 300

Termination Policies Default

Creation Time Thu Nov 30 19:41:43 GMT-800 2017

Availability Zone(s) us-west-1b, us-west-1c
Subnet(s) subnet-12f63f75, subnet-994ebc1
Default Cooldown 300
Placement Group
Suspended Processes
Enabled Metrics
Instance Protection

Feedback English (US) © 2008-2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Shopping cart – Load Balancer & Auto Scaling Group

AWS Services Resource Groups

EC2 Dashboard

Events

Tags

Reports

Limits

INSTANCES

Instances

Launch Templates

Spot Requests

Reserved Instances

Dedicated Hosts

IMAGES

AMIs

Bundle Tasks

ELASTIC BLOCK STORE

Volumes

Snapshots

NETWORK & SECURITY

Security Groups

Elastic IPs

Placement Groups

Key Pairs

Launch Instance Connect Actions

Filter by tags and attributes or search by keyword

1 to 5 of 5

Name	Instance ID	Instance Type	Availability Zone	Instance Status	Status Checks	Alarm	Public DNS
order-payment-autoscale-server	i-0dcdeba78857e16e3	t2.micro	us-west-1b	running	2/2 checks	ec2-54-67-85
OrderPaymentServer	i-01add534a82369e98	t2.micro	us-west-1b	running	2/2 checks	ec2-54-193-6
MongoDB-Server-3	i-06808a7b35c1c616e	t2.micro	us-west-1c	running	2/2 checks	ec2-54-183-2
MongoDB-Server-2	i-076485acb7c54fdfc	t2.micro	us-west-1b	running	2/2 checks	ec2-54-215-1
MongoDB-Server-1	i-0842244b789059d69	t2.micro	us-west-1b	running	2/2 checks	ec2-54-183-2

Select an instance above

Feedback English (US) © 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

AWS Services Resource Groups

Reserved Instances

Dedicated Hosts

IMAGES

AMIs

Bundle Tasks

ELASTIC BLOCK STORE

Volumes

Snapshots

NETWORK & SECURITY

Security Groups

Elastic IPs

Placement Groups

Key Pairs

Network Interfaces

LOAD BALANCING

Load Balancers

Target Groups

AUTO SCALING

Launch Configurations

Auto Scaling Groups

SYSTEMS MANAGER SERVICES

Create Security Group Actions

Filter by tags and attributes or search by keyword

1 to 3 of 3

Name	Group ID	Group Name	VPC ID	Description
sg-81a29fe7	mongodb-cluster	vpc-d40415b0	Security group for MongoDB cluster	
sg-9e8080f8	api-group	vpc-d40415b0	This security group is to be used for API la...	
sg-d49ca1b2	default	vpc-d40415b0	default VPC security group	

Type	Protocol	Port Range	Source	Description
Custom TCP Rule	TCP	27019	0.0.0.0/0	
Custom TCP Rule	TCP	27019	::/0	
Custom TCP Rule	TCP	27018	0.0.0.0/0	
Custom TCP Rule	TCP	27018	::/0	
SSH	TCP	22	0.0.0.0/0	
SSH	TCP	22	::/0	
Custom TCP Rule	TCP	5000	0.0.0.0/0	
Custom TCP Rule	TCP	5000	::/0	
Custom TCP Rule	TCP	4000	0.0.0.0/0	
Custom TCP Rule	TCP	4000	::/0	

Feedback English (US) © 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Orders/Payments– Security Group & EC2

EC2 Management Console

Secure | https://us-west-1.console.aws.amazon.com/ec2/v2/home?region=us-west-1#LoadBalancers:

Aashish Subramanian N. California Support

Create Load Balancer Actions

Reserved Instances
Dedicated Hosts

IMAGES
AMIs
Bundle Tasks

ELASTIC BLOCK STORE
Volumes
Snapshots

NETWORK & SECURITY
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces

LOAD BALANCING
Load Balancers Target Groups

AUTO SCALING Launch Configurations Auto Scaling Groups

SYSTEMS MANAGER SERVICES

Filter: Search 1 to 1 of 1

Name	DNS name	State	VPC ID	Availability Zones	Type
cmpe281-project-balancer	cmpe281-project-balancer-1...	InService	vpc-d40415b0	us-west-1b, us-west-1c	classic

Description Instances Health Check Listeners Monitoring Tags Migration

Connection Draining: Enabled, 300 seconds (Edit)

Edit Instances

Instance ID	Name	Availability Zone	Status	Actions
i-01add534a82369e98	OrderPaymentServer	us-west-1b	InService	Remove from Load Balancer
i-0dcedba78857e16e3	order-payment-autoscale-server	us-west-1b	InService	Remove from Load Balancer

Edit Availability Zones

Availability Zone	Subnet ID	Subnet CIDR	Instance Count	Healthy?	Actions
us-west-1b	subnet-1a955641	10.0.0.0/24	2	Yes	Remove from Load Balancer
us-west-1c	subnet-8404b7e3	10.0.2.0/24	0	No (Availability Zone contains no healthy targets)	Remove from Load Balancer

Feedback English (US) © 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

EC2 Management Console

Secure | https://us-west-1.console.aws.amazon.com/ec2/autoscaling/home?region=us-west-1#LaunchConfigurations:id=cmpe281-project

Aashish Subramanian N. California Support

Create launch configuration Create Auto Scaling group Actions

EC2 Dashboard Events Tags Reports Limits

INSTANCES Instances Launch Templates Spot Requests Reserved Instances Dedicated Hosts

IMAGES AMIs Bundle Tasks

ELASTIC BLOCK STORE Volumes Snapshots

NETWORK & SECURITY Security Groups Elastic IPs Placement Groups Key Pairs Network Interfaces

LOAD BALANCING

Launch Configuration: cmpe281-project

Details

AMI ID	ami-9a566cfa	Instance Type	t2.micro
IAM Instance Profile	cmpe281-project	Kernel ID	
Key Name	cmpe281-project	Monitoring	false
EBS Optimized	false	Security Groups	sg-81a29fe7
Spot Price		Creation Time	Fri Dec 01 02:00:46 GMT-800 2017
RAM Disk ID		Block Devices	/dev/xvda
User data	View User data	IP Address Type	Assign a public IP address to every instance.

Copy launch configuration

Feedback English (US) © 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Orders/Payments-Load Balancer & Auto Scaling Group

Secure | <https://us-west-1.console.aws.amazon.com/ec2/v2/home?region=us-west-1#InstancesInstanceState=running;sort=publicIp>

Services Resource Groups Actions

Launch Instance Connect Actions

Instance State : Running Add filter

1 to 5 of 5

Name	Instance ID	Availability Zone	Instance State	Public DNS (IPv4)	IPv4 Public IP
a_mongo2	i-0d441101aadfaaf6	us-west-1b	running	ec2-13-56-58-166.us-west-1.compute.amazonaws.com	13.56.58.166
logSystemBackend_two	i-0ee122a799ff35d	us-west-1b	running	ec2-13-56-184-83.us-west-1.compute.amazonaws.com	13.56.184.83
logSystemBackend_one	i-03fdb43f006597edb	us-west-1b	running	ec2-54-183-181-217.us-west-1.compute.amazonaws.com	54.183.181.217
a_mongo3	i-0b78ebb937ca93233	us-west-1b	running	ec2-54-215-228-194.us-west-1.compute.amazonaws.com	54.215.228.194
a_mongo1	i-073255b22cc24036	us-west-1b	running	ec2-54-241-150-45.us-west-1.compute.amazonaws.com	54.241.150.45

Instance: i-03fdb43f006597edb (logSystemBackend_one) Public DNS: ec2-54-183-181-217.us-west-1.compute.amazonaws.com

Description Status Checks Monitoring Tags

Instance ID: i-03fdb43f006597edb Public DNS (IPv4): ec2-54-183-181-217.us-west-1.compute.amazonaws.com

Instance state: running IPv4 Public IP: 54.183.181.217

Instance type: t2.micro IPv6 IPs: -

Elastic IPs: Private DNS: ip-10-0-0-32.us-west-1.compute.internal

Availability zone: us-west-1b Private IPs: 10.0.0.32

Security groups: nodejsSG, view inbound rules Secondary private IPs:

Scheduled events: No scheduled events VDC ID: vnc-273e2053

<https://us-west-1.console.aws.amazon.com/ec2/v2/home?region=us-west-1#SecurityGroups:sort=groupId>

Services Resource Groups Actions

Create Security Group Actions

Filter by tags and attributes or search by keyword

1 to 10 of 10

Name	Group ID	Group Name	VPC ID	Description
sg-26e67640	default	vpc-73550d17	default VPC security group	
sg-76ba3e10	Loadbalancer SG	vpc-2f42144b	inbound/Outbound connections	
sg-7a3d3d1c	mongo-cluster	vpc-373e2053	launch-wizard-1 created 2017-11-29T22:49:54.11...	
sg-8bbb3fed	Web-Tier SG	vpc-2f42144b	Webtier traffic	
sg-9f3232f9	mongodb-cluster	vpc-2f42144b	SG for mongo	
sg-a08e1ec6	dc-vpc-default	vpc-73550d17	Default VPC security group for Docker Cloud nodes	
sg-a2cecccc4	nodejsSG	vpc-373e2053	SG for Nodejs apps.	
sg-b2ba3ed4	BackendTier SG	vpc-2f42144b	Group for the backend DB	

Security Group: sg-a2cecccc4

Description Inbound Outbound Tags

Group name: nodejsSG Group description: SG for Nodejs apps.

Group ID: sg-a2cecccc4 VPC ID: vpc-373e2053

LogSystem-Security Group and EC2 instances

services Resource Groups

Create Load Balancer Actions

Filter: Search 1 to 1 of 1

Name	DNS name	State	VPC ID	Availability Zones	Type
log-api-LoadBalancer	log-api-LoadBalancer-17386...	Active	vpc-373e2053	us-west-1b	classic

Load balancer: log-api-LoadBalancer

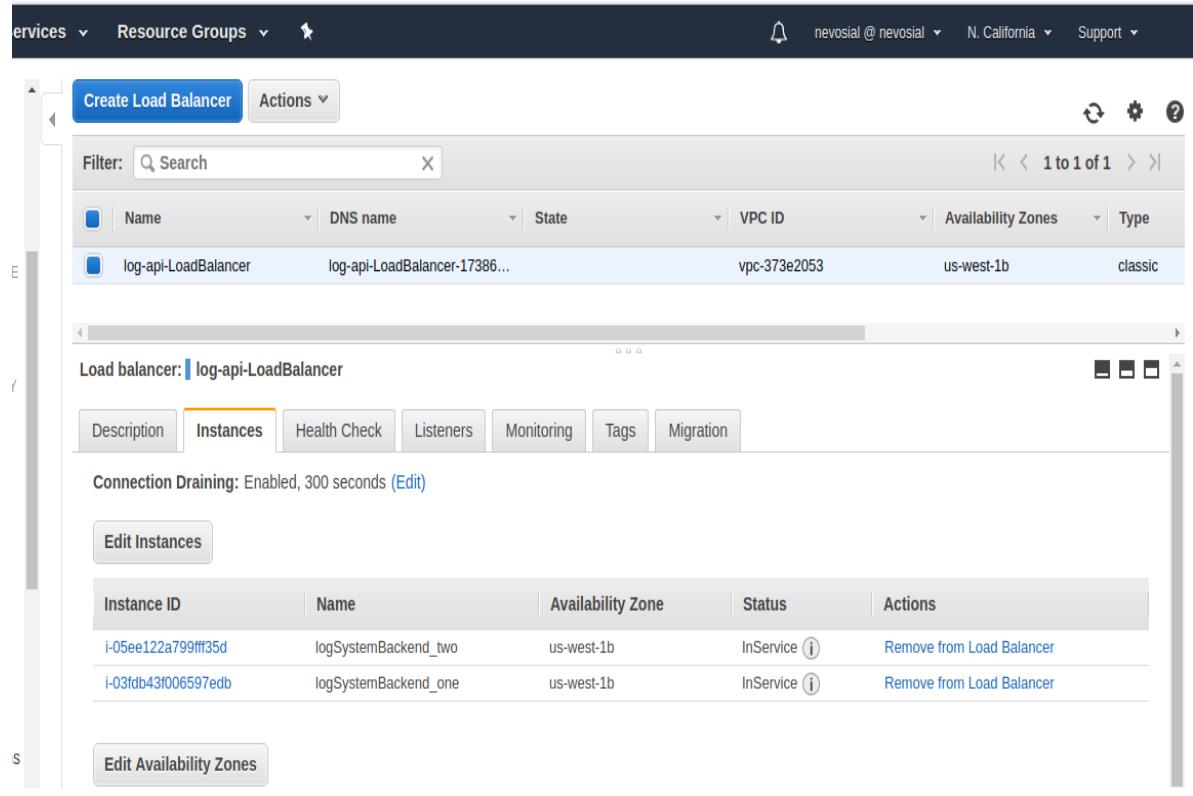
Description Instances Health Check Listeners Monitoring Tags Migration

Connection Draining: Enabled, 300 seconds (Edit)

Edit Instances

Instance ID	Name	Availability Zone	Status	Actions
i-05ee122a799ff35d	logSystemBackend_two	us-west-1b	InService	Remove from Load Balancer
i-03fdb43f006597edb	logSystemBackend_one	us-west-1b	InService	Remove from Load Balancer

Edit Availability Zones



https://us-west-1.console.aws.amazon.com/ec2/autoscaling/home?region=us-west-1#AutoScalingGroups:id=Log_backend_apis;view... 11

Create Auto Scaling group Actions

Filter: Filter Auto Scaling groups... 1 to 1 of 1 Auto Scaling Groups

Name	Launch Configuration /	Instances	Desired	Min	Max	Availability Zones	Default Cooldown	Health
Log_backend_apis	log-backendapi-servers	1	1	1	2	us-west-1a, us-west-1b	300	300

Launch Configuration: log-backendapi-servers

Launch Template: log-backendapi-servers

Launch Template Version: 1

Load Balancers: log-api-LoadBalancer

Target Groups:

Desired	Min	Max
1	1	2

Health Check Type: EC2

Health Check Grace Period: 300

Termination Policies: Default

Creation Time: Fri Dec 01 21:12:30 GMT 2017

Availability Zone(s): us-west-1a, us-west-1b

Subnet(s): subnet-74bd0513, subnet-ab18d3f0

Default Cooldown: 300

Placement Group:

Suspended Processes:

Enabled Metrics:

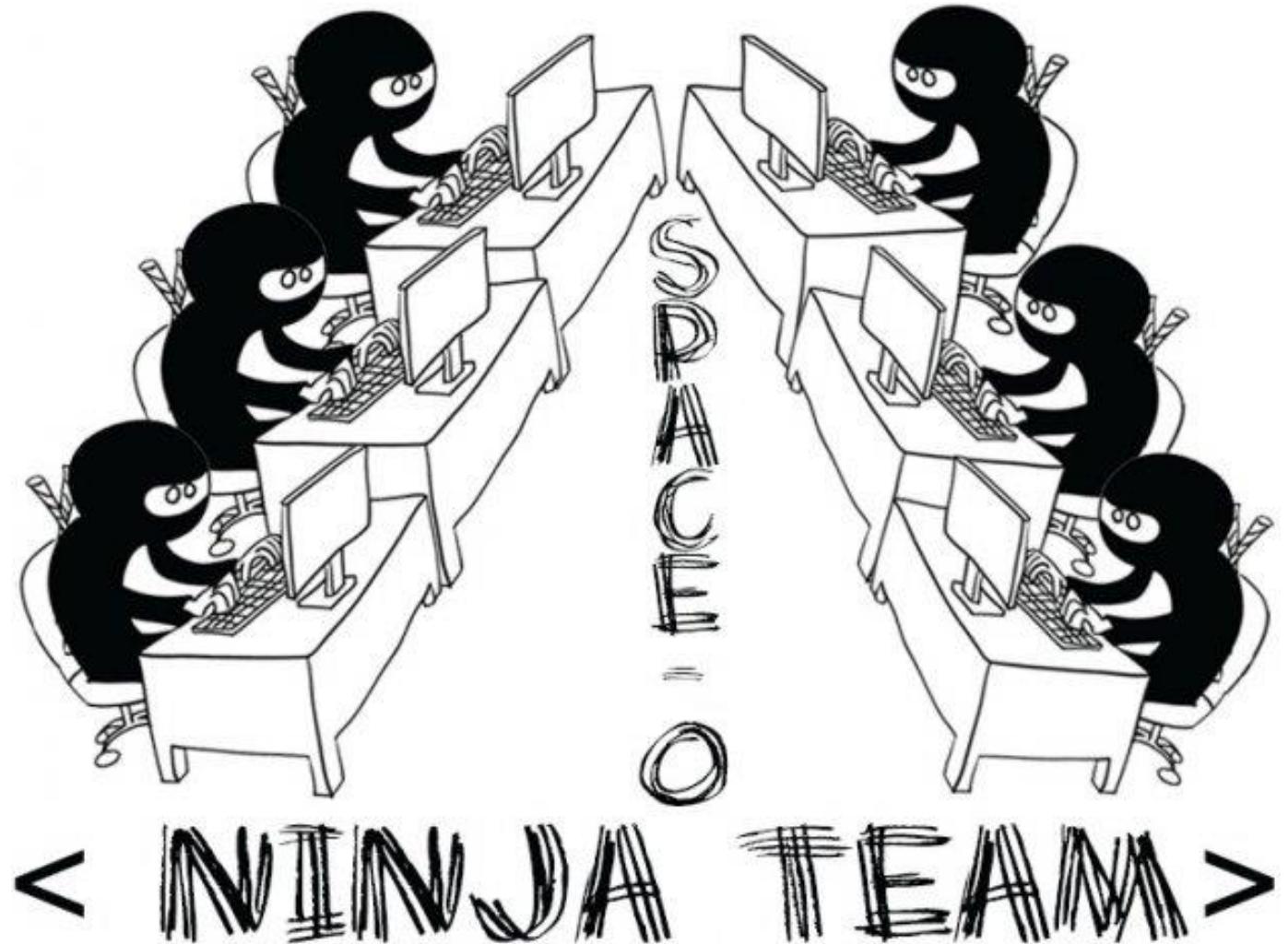
Instance Protection:

LogSystem -Load Balancer & Auto Scaling Group

<link>

Demo

Team Retrospective



What did we do well?

- Finalized the architecture early.
- Weekly team meetings.
- Always followed the core values.
- Regular feedback from team members.
- Focused on what was required.
- The tasks were completed weekly and on time.
- The task for next week were finalized a week before so that one can get ample time to complete it.

What needs improvement?

- Start the coding part as early as possible.
- Testing should be done right from the start.
- End to End testing is necessary.
- Meetings should be more regular like 3 times in a week.

Team Retrospective

What should we keep doing?

- Focus on the main objective of the project.
- Have regular team meeting.
- Complete the task as per the schedule.
- Help others if they are stuck.
- Communicate on regular basis.

What went wrong during the sprint cycle?

- Setting unrealistic deadlines.
- Testing only individual modules.
- Integrating the whole project in the last phase.

Team Retrospective



THANK
YOU