

Senior Thesis Progress

December 20, 2018

Outline

- ▶ Problem Statement

Outline

- ▶ Problem Statement
 - ▶ Background of Phenomena and Technologies
 - ▶ Detailed Problem Statement

Outline

- ▶ Problem Statement
 - ▶ Background of Phenomena and Technologies
 - ▶ Detailed Problem Statement
- ▶ Current Progress

Outline

- ▶ Problem Statement
 - ▶ Background of Phenomena and Technologies
 - ▶ Detailed Problem Statement
- ▶ Current Progress
 - ▶ Month by month breakdown.
 - ▶ Position with respect to milestones

Outline

- ▶ Problem Statement
 - ▶ Background of Phenomena and Technologies
 - ▶ Detailed Problem Statement
- ▶ Current Progress
 - ▶ Month by month breakdown.
 - ▶ Position with respect to milestones
- ▶ Remaining Work

Outline

- ▶ Problem Statement
 - ▶ Background of Phenomena and Technologies
 - ▶ Detailed Problem Statement
- ▶ Current Progress
 - ▶ Month by month breakdown.
 - ▶ Position with respect to milestones
- ▶ Remaining Work
 - ▶ Technical Details
 - ▶ Execution Plan

Problem Statement

TLDR

Make a dishwasher autonomously turn on when electricity is cheapest.

Background

IOT

Intelligently interact with everyday objects over a network.

Particle



Background

IOT

Intelligently interact with everyday objects over a network.

Background

IOT

Intelligently interact with everyday objects over a network.

Serverless Computing

Outsource the maintenance of a server to a third party. Focus on the product.

Example of Serverless Computing

- ▶ Kellogg's spends $1/3$ of its revenue on trade spend.

Example of Serverless Computing

- ▶ Kellogg's spends $1/3$ of its revenue on trade spend.
 - ▶ Optimize trade spend over choices:
 - ▶ Coupons
 - ▶ Ad campaigns
 - ▶ TV ad spend
 - ▶ Shelving and Display

Example of Serverless Computing

- ▶ Kellog's spends 1/3 of its revenue on trade spend.
 - ▶ Optimize trade spend over choices:
 - ▶ Coupons
 - ▶ Ad campaigns
 - ▶ TV ad spend
 - ▶ Shelving and Display
- ▶ Until 2013, Kellog's ran queries on an on-premise, relational database.

Example of Serverless Computing

- ▶ Kellogg's spends 1/3 of its revenue on trade spend.
 - ▶ Optimize trade spend over choices:
 - ▶ Coupons
 - ▶ Ad campaigns
 - ▶ TV ad spend
 - ▶ Shelving and Display
- ▶ Until 2013, Kellogg's ran queries on an on-premise, relational database.
- ▶ Reached 16 TB data

Example of Serverless Computing

- ▶ Kellogg's spends 1/3 of its revenue on trade spend.
 - ▶ Optimize trade spend over choices:
 - ▶ Coupons
 - ▶ Ad campaigns
 - ▶ TV ad spend
 - ▶ Shelving and Display
- ▶ Until 2013, Kellogg's ran queries on an on-premise, relational database.
- ▶ Reached 16 TB data
 - ▶ They outsourced the infrastructure needed to run and store these queries to AWS (Amazon Web Services).

Background

IOT

Intelligently interact with everyday objects over a network.

Serverless Computing

Outsource the maintenance of a server to a third party. Focus on the product.

Background

IOT

Intelligently interact with everyday objects over a network.

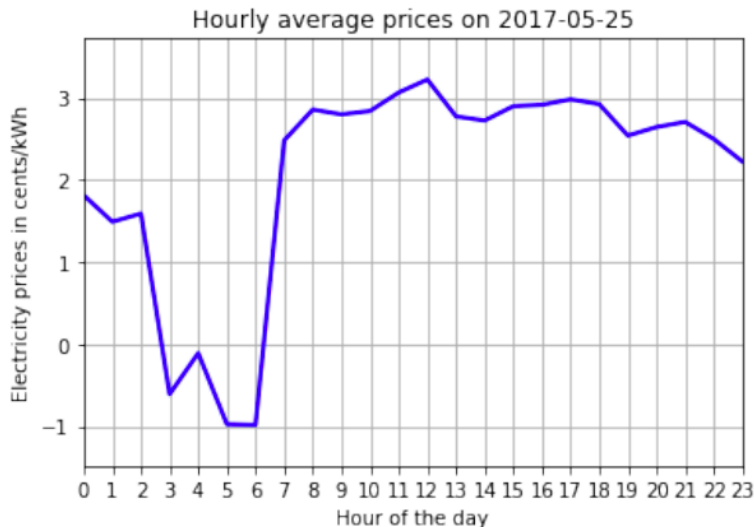
Serverless Computing

Outsource the maintenance of a server to a third party. Focus on the product.

Electricity Price Variability

Predict when rates are the cheapest.

Price Data from ComEd in Illinois



Background

IOT

Intelligently interact with everyday objects over a network.

Serverless Computing

Outsource the maintenance of a server to a third party. Focus on the product.

Electricity Price Variability

Predict when rates are the cheapest. [6]

Background

IOT

Intelligently interact with everyday objects over a network.

Serverless Computing

Outsource the maintenance of a server to a third party. Focus on the product.

Electricity Price Variability

Predict when rates are the cheapest. [6]

Environmental Appeal

There is high correlation between prices are cheap and low grid demand, which is in turn correlated with more environmentally friendly uses of power [5] [3].

Full Problem

- ▶ Frame a heuristic in place of identifying when electricity prices are cheapest.

Full Problem

- ▶ Frame a heuristic in place of identifying when electricity prices are cheapest.
- ▶ Make an algorithm to solve that heuristic.

Algorithm

- ▶ Run the algorithm every hour from 12AM to 5AM until the algorithm says “yes.”

Algorithm

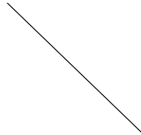
- ▶ Run the algorithm every hour from 12AM to 5AM until the algorithm says “yes.”
- ▶ Classify between 3AM and 4AM.

Full Problem

- ▶ Frame a heuristic in place of identifying when electricity prices are cheapest.
- ▶ Make an algorithm to solve that heuristic.

Full Problem

- ▶ Frame a heuristic in place of identifying when electricity prices are cheapest.
- ▶ Make an algorithm to solve that heuristic.
- ▶ Make a subscription service that allows consumers to enable their electricity appliances to turn on at later, cheaper times.



Full Problem

- ▶ Frame a heuristic in place of identifying when electricity prices are cheapest.
- ▶ Make an algorithm to solve that heuristic.
- ▶ Make a subscription service that allows consumers to enable their electricity appliances to turn on at later, cheaper times.

Full Problem

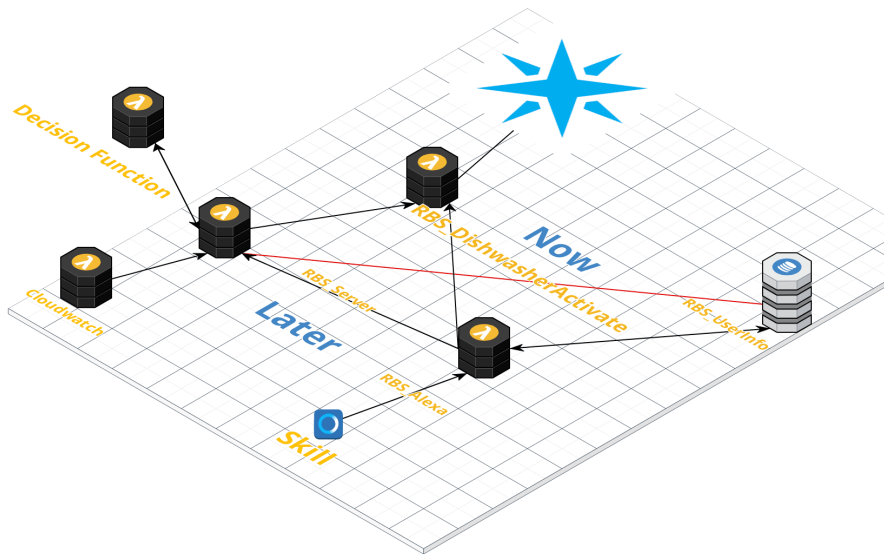
- ▶ Frame a heuristic in place of identifying when electricity prices are cheapest.
- ▶ Make an algorithm to solve that heuristic.
- ▶ Make a subscription service that allows consumers to enable their electricity appliances to turn on at later, cheaper times.
 - ▶ Does this service scale well when used with many clients?
What are the results from a heavy simulation?

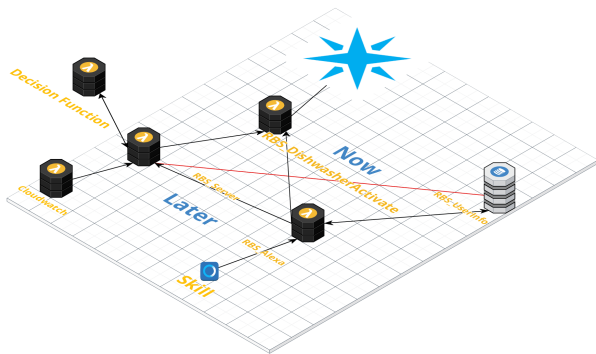
Full Problem

- ▶ Frame a heuristic in place of identifying when electricity prices are cheapest.
- ▶ Make an algorithm to solve that heuristic.
- ▶ Make a subscription service that allows consumers to enable their electricity appliances to turn on at later, cheaper times.
 - ▶ Does this service scale well when used with many clients? What are the results from a heavy simulation?
 - ▶ Does this service actually save money when physically implemented [4]? Can we, as service providers, make the service profitable?

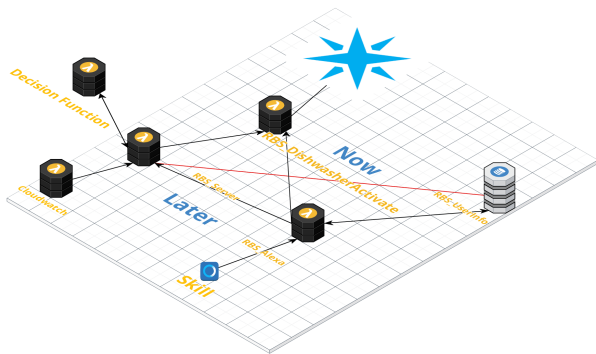
Full Problem

- ▶ Frame a heuristic in place of identifying when electricity prices are cheapest.
- ▶ Make an algorithm to solve that heuristic.
- ▶ Make a subscription service that allows consumers to enable their electricity appliances to turn on at later, cheaper times.
 - ▶ Does this service scale well when used with many clients? What are the results from a heavy simulation?
 - ▶ Does this service actually save money when physically implemented [4]? Can we, as service providers, make the service profitable?
 - ▶ If time permits: If such a service begins to be used en masse, what outlook does the profitability of it have? Is there still a profit that can be made?

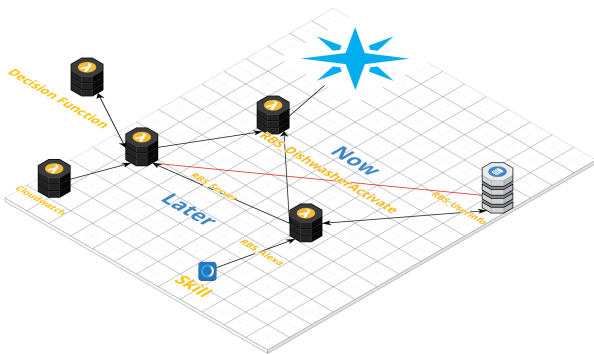




► Problems:

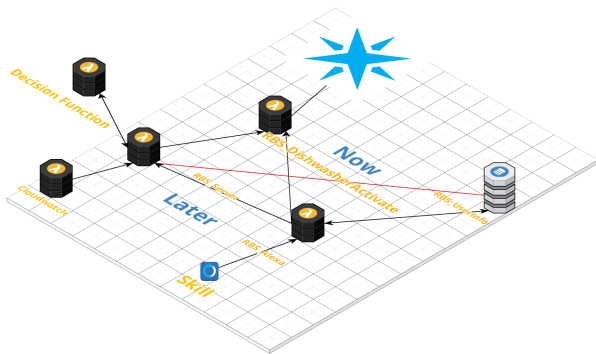


- Problems:
 - Not scaled.



► Problems:

- Not scaled.
- Interaction with algorithm does not work – there is no way of testing its success.



► Problems:

- Not scaled.
- Interaction with algorithm does not work – there is no way of testing its success.
- Interaction with Alexa is unidirectional.

September

S	M	T	W	T	F	S
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

- Learned how to work with:

September

S	M	T	W	T	F	S
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

- ▶ Learned how to work with:
 - ▶ AWS
 - ▶ Lambda
 - ▶ DynamoDB
 - ▶ Cloudwatch

September

S	M	T	W	T	F	S
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

- ▶ Learned how to work with:
 - ▶ AWS
 - ▶ Lambda
 - ▶ DynamoDB
 - ▶ Cloudwatch
 - ▶ Javascript and Node

September

S	M	T	W	T	F	S
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

- ▶ Learned how to work with:
 - ▶ AWS
 - ▶ Lambda
 - ▶ DynamoDB
 - ▶ Cloudwatch
 - ▶ Javascript and Node
 - ▶ Particle

October

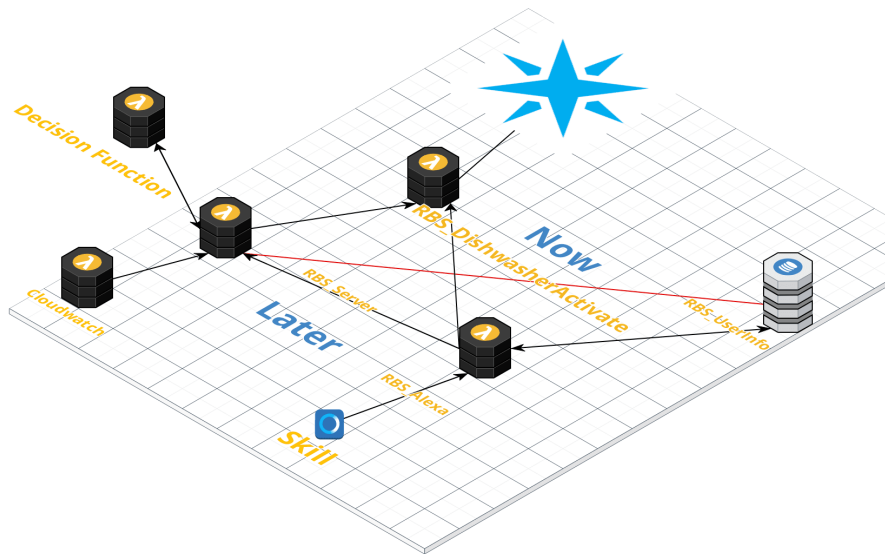
S	M	T	W	T	F	S
30	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

- ▶ First two weeks:
 - ▶ Revived Old Infrastructure

October

S	M	T	W	T	F	S
30	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

- ▶ First two weeks:
 - ▶ Revived Old Infrastructure
- ▶ Second two weeks:
 - ▶ Scaled the existing infrastructure to work with two appliances for one user when invoking the appliance immediately.



October

S	M	T	W	T	F	S
30	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

- ▶ First two weeks:
 - ▶ Revived Old Infrastructure
- ▶ Second two weeks:
 - ▶ Scaled the existing infrastructure to work with two appliances for one user when invoking the appliance immediately.

October

S	M	T	W	T	F	S
30	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

- ▶ First two weeks:
 - ▶ Revived Old Infrastructure
- ▶ Second two weeks:
 - ▶ Scaled the existing infrastructure to work with two appliances for one user when invoking the appliance immediately.
- ▶ Noticed problem in workflow.

Workflow Challenges

- ▶ By its very nature, serverless computing is a black box [2]:

Workflow Challenges

- ▶ By its very nature, serverless computing is a black box [2]:
 - ▶ Inability to work locally:

Workflow Challenges

- ▶ By its very nature, serverless computing is a black box [2]:
 - ▶ Inability to work locally:
 - ▶ Limits version control.

Workflow Challenges

- ▶ By its very nature, serverless computing is a black box [2]:
 - ▶ Inability to work locally:
 - ▶ Limits version control.
 - ▶ Limits teamwork.

Workflow Challenges

- ▶ By its very nature, serverless computing is a black box [2]:
 - ▶ Inability to work locally:
 - ▶ Limits version control.
 - ▶ Limits teamwork.
 - ▶ Limits programming tool usage (ie debuggers and editor customization)

Workflow Challenges

- ▶ By its very nature, serverless computing is a black box [2]:
 - ▶ Inability to work locally:
 - ▶ Limits version control.
 - ▶ Limits teamwork.
 - ▶ Limits programming tool usage (ie debuggers and editor customization)
 - ▶ Inability to test:

Workflow Challenges

- ▶ By its very nature, serverless computing is a black box [2]:
 - ▶ Inability to work locally:
 - ▶ Limits version control.
 - ▶ Limits teamwork.
 - ▶ Limits programming tool usage (ie debuggers and editor customization)
 - ▶ Inability to test:
 - ▶ Introduces bugs into existing code.

Workflow Challenges

- ▶ By its very nature, serverless computing is a black box [2]:
 - ▶ Inability to work locally:
 - ▶ Limits version control.
 - ▶ Limits teamwork.
 - ▶ Limits programming tool usage (ie debuggers and editor customization)
 - ▶ Inability to test:
 - ▶ Introduces bugs into existing code.
 - ▶ Leads to longer development time.

Workflow Challenges

- ▶ By its very nature, serverless computing is a black box [2]:
 - ▶ Inability to work locally:
 - ▶ Limits version control.
 - ▶ Limits teamwork.
 - ▶ Limits programming tool usage (ie debuggers and editor customization)
 - ▶ Inability to test:
 - ▶ Introduces bugs into existing code.
 - ▶ Leads to longer development time.
- ▶ Solution: Devops

Devops Developments

November

S	M	T	W	T	F	S
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	1
2	3	4	5	6	7	8

December

S	M	T	W	T	F	S
25	26	27	28	29	30	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

- ▶ A script that allows for remote execution and uploading of lambda function code.
 - ▶ First two weeks of November.

Devops Developments

November

S	M	T	W	T	F	S
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	1
2	3	4	5	6	7	8

December

S	M	T	W	T	F	S
25	26	27	28	29	30	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

- ▶ A script that allows for remote execution and uploading of lambda function code.
 - ▶ First two weeks of November.
- ▶ Infrastructure as Code and learned common testing frameworks.
 - ▶ Second two weeks of November.

Devops Developments

November

S	M	T	W	T	F	S
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	1
2	3	4	5	6	7	8

December

S	M	T	W	T	F	S
25	26	27	28	29	30	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

- ▶ A script that allows for remote execution and uploading of lambda function code.
 - ▶ First two weeks of November.
- ▶ Infrastructure as Code and learned common testing frameworks.
 - ▶ Second two weeks of November.
- ▶ Testing frameworks and tests for Alexa Code and Lambda Function Code
 - ▶ December

Devops Developments

November

S	M	T	W	T	F	S
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	1
2	3	4	5	6	7	8

December

S	M	T	W	T	F	S
25	26	27	28	29	30	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

- ▶ A script that allows for remote execution and uploading of lambda function code.
 - ▶ First two weeks of November.
- ▶ Infrastructure as Code and learned common testing frameworks.
 - ▶ Second two weeks of November.
- ▶ Testing frameworks and tests for Alexa Code and Lambda Function Code
 - ▶ December
- ▶ Using these tools, I've made, thus far, a few slight changes.

Progress with Respect to Milestones

▶ December 2018

- ▶ Document half of the repository code. 25%
- ▶ ~~Devise a local workflow for the development of the Alexa UI.~~
- ▶ Refactor and test a function called ~~RBS_Lambda~~ RBS_dishwasher_activate. 100%
- ▶ Have a technical report of the prediction models tried by the group up until now. 0%

▶ November 2018

- ▶ Devise a local workflow for AWS lambda. At present, we use AWS's interface for most of our programming. We need a local workflow to streamline testing and development. 100%
- ▶ ~~Refactor and test a function called RBS_argmin.~~
- ▶ Refactor and test a function called RBS_server. 75%
- ▶ Refactor and test a function called RBS_Alexa 75%

Future Work

- ▶ Devise a way of remotely verifying whether the infrastructure caused the device to switch on at an “optimal” time.

Future Work

- ▶ Devise a way of remotely verifying whether the infrastructure caused the device to switch on at an “optimal” time.
- ▶ Devise a way to simulate many clients using this service and to test the ability of the service to handle their requests.

Future Work

- ▶ Devise a way of remotely verifying whether the infrastructure caused the device to switch on at an “optimal” time.
- ▶ Devise a way to simulate many clients using this service and to test the ability of the service to handle their requests.
- ▶ Expand the range of utterances the Alexa front end can use.

Future Work

- ▶ Devise a way of remotely verifying whether the infrastructure caused the device to switch on at an “optimal” time.
- ▶ Devise a way to simulate many clients using this service and to test the ability of the service to handle their requests.
- ▶ Expand the range of utterances the Alexa front end can use.
- ▶ Make the dialogues interactive.

Future Work

- ▶ Devise a way of remotely verifying whether the infrastructure caused the device to switch on at an “optimal” time.
- ▶ Devise a way to simulate many clients using this service and to test the ability of the service to handle their requests.
- ▶ Expand the range of utterances the Alexa front end can use.
- ▶ Make the dialogues interactive.
- ▶ Change the database schema to remove duplication of appliance information.

Future Work

- ▶ Devise a way of remotely verifying whether the infrastructure caused the device to switch on at an “optimal” time.
- ▶ Devise a way to simulate many clients using this service and to test the ability of the service to handle their requests.
- ▶ Expand the range of utterances the Alexa front end can use.
- ▶ Make the dialogues interactive.
- ▶ Change the database schema to remove duplication of appliance information.
- ▶ Refactor RBS_server to use multiple devices.

Works Cited I



S. Reed.

Power prices go negative in germany, a positive for energy users

New York Times



Baldini

Serverless computing: Current trends and open problems

Research Advances in Cloud Computing



Lombard

A review on buildings energy consumption information.

Energy and Buildings



Faruqui

The impact of informational feedback on energy consumption—A survey of the experimental evidence

Energy

Works Cited II



Mooney

There's a big change coming to how we power our homes —
and it isn't about solar or batteries

Washington Post



Sowers

A smart power outlet for electric devices that can benefit from
Real-Time Pricing

*2017 International Conference on Control, Electronics,
Renewable Energy and Communications (ICCREC)*