

Online Home Automation Control System

Graduation Project 1 Presentation
Group 45 - JC7
Dr Abeer Mahmoud

Introduction

- **Very fast increase in automation** in the recent decade
- **Higher demand** for applications that can support remote control
- especially electrically controlled house appliances



Problem Statement

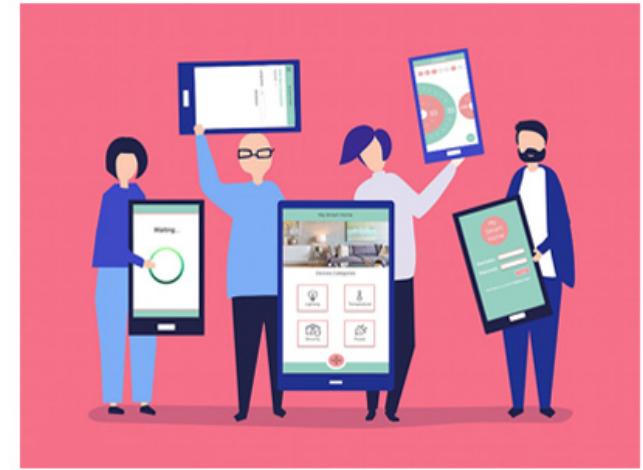
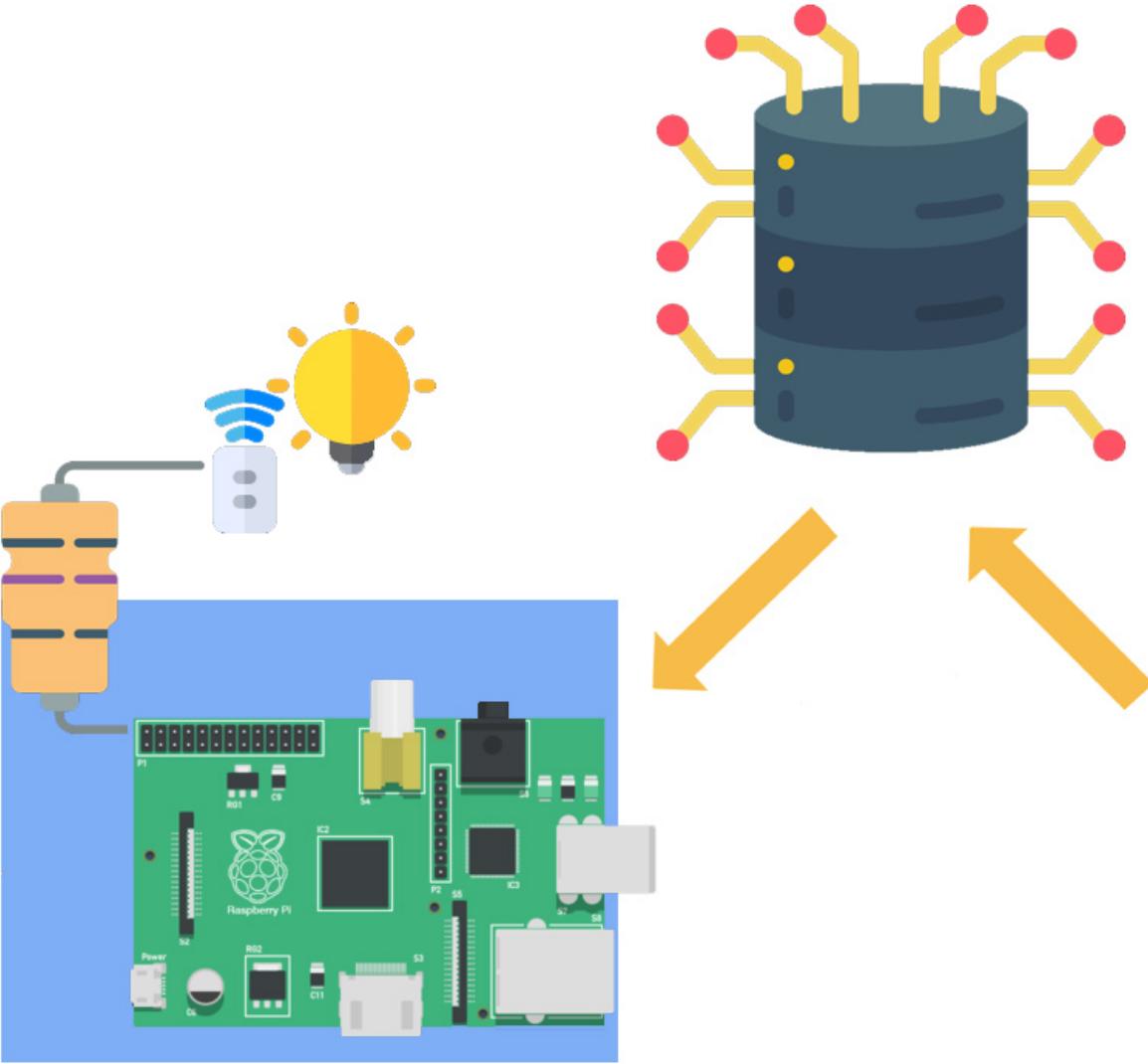
- **Many applications available** providing remote control at home
- **All work in a small set of devices and sensors**
- Main challenge: control these devices through the Internet
- Important for pet owners with UV lights, parents waking their children up etc



Objective and Proposed Solution

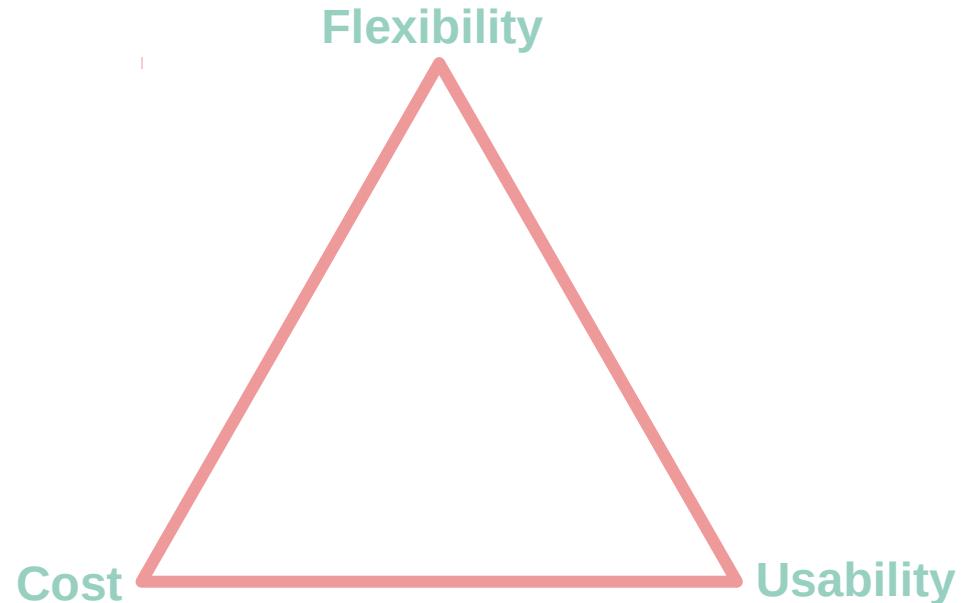
- **Objective:**
 - remote control of house appliances through IoT
- **Solution:**
 - creating **Android mobile** application that controls switches/lights etc by issuing commands
 - using **Raspberry Pi** to switch the lights/switches on and off
 - creating a **web application** interface between the mobile application and laptop



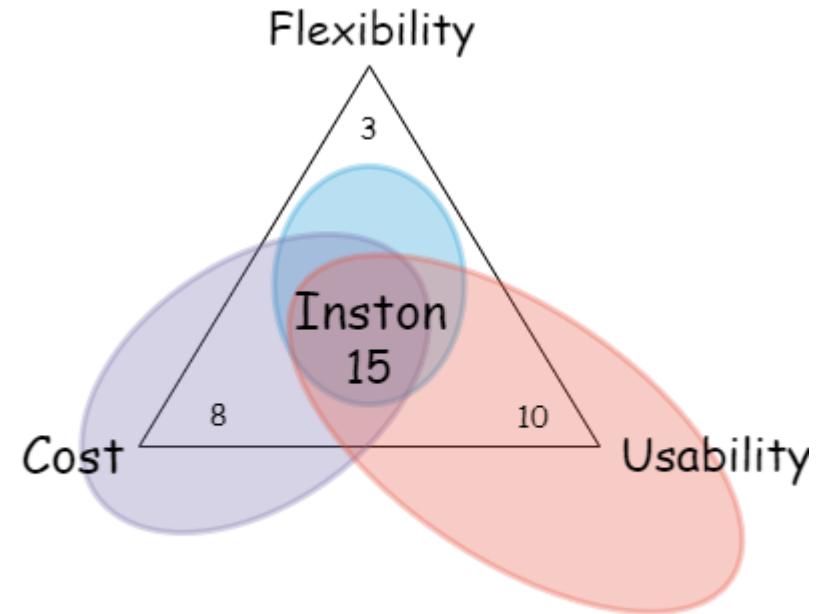


Related Work

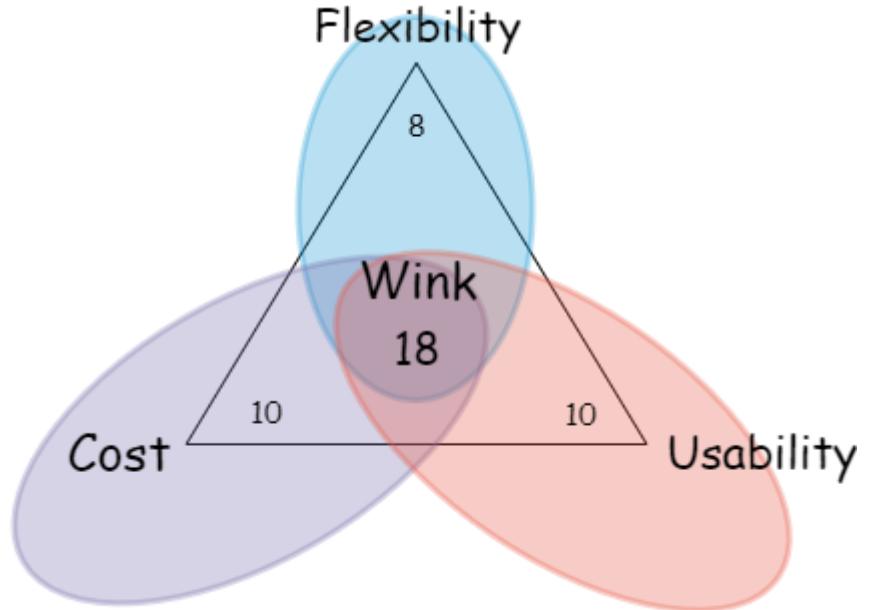
- Some examples of similar applications:
 - Insteon
 - Wink hub 2
 - Samsung smart things



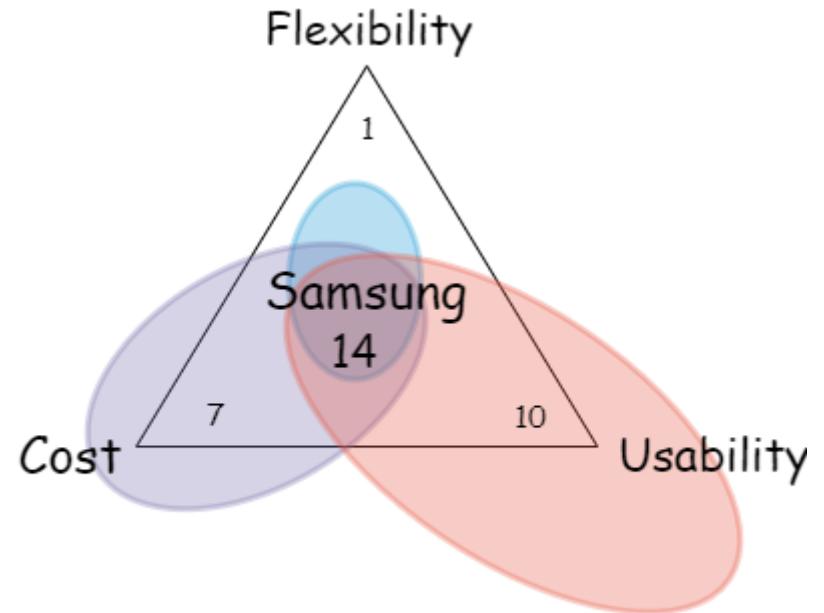
Related Work - Insteon



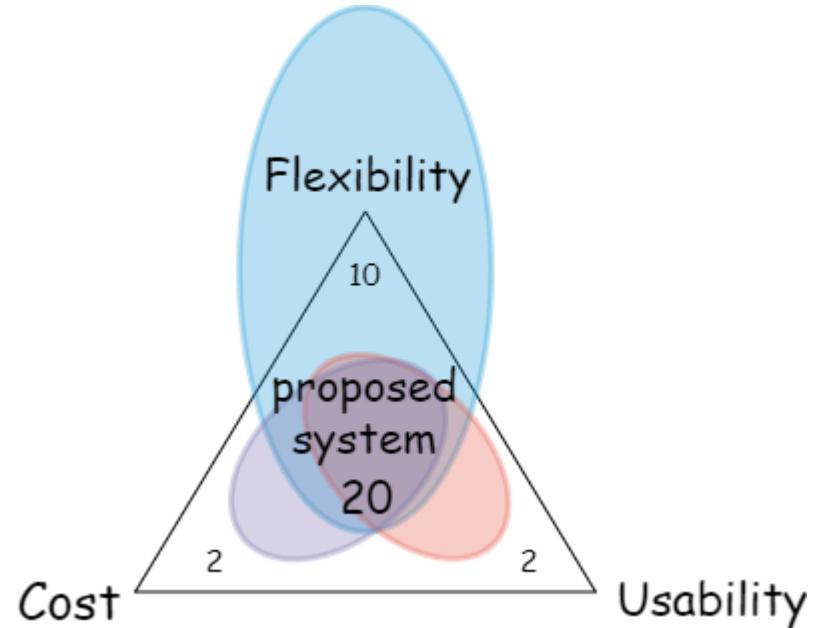
Related Work - Wink



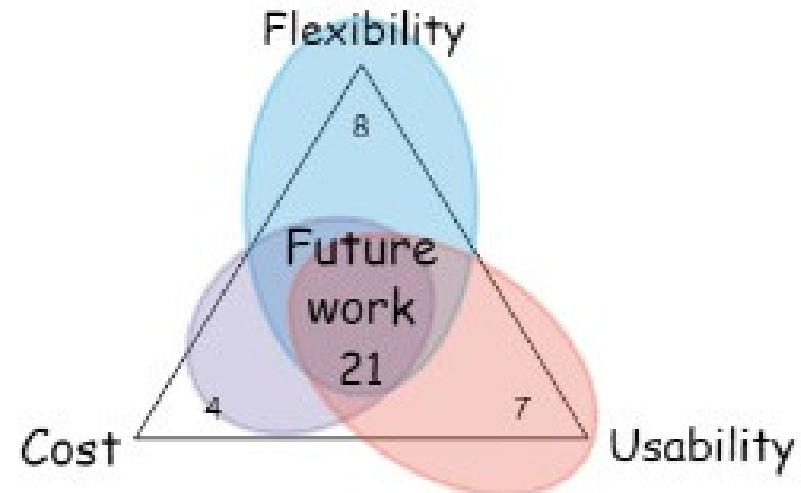
Related Work - Samsung



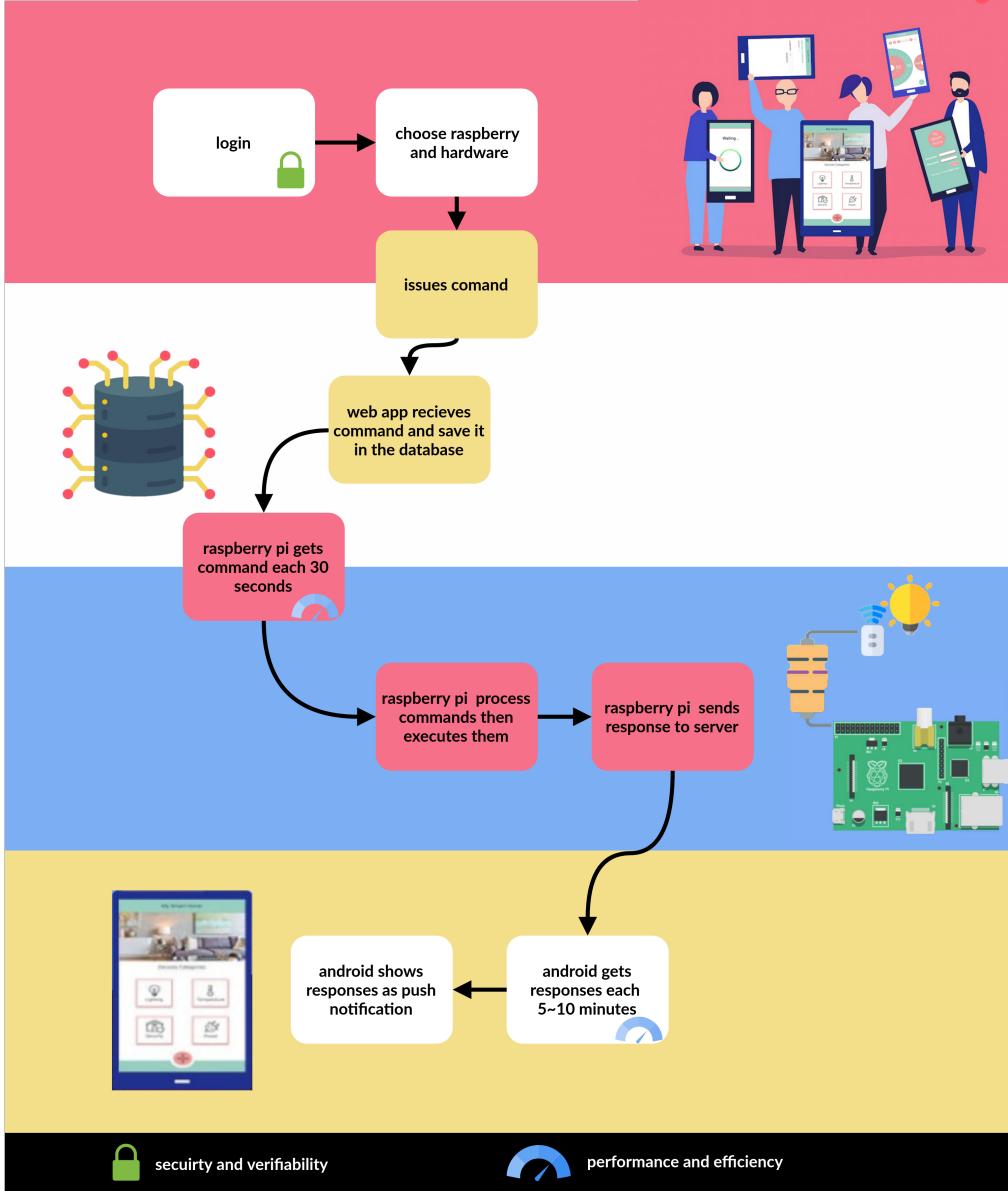
Related Work - Proposed

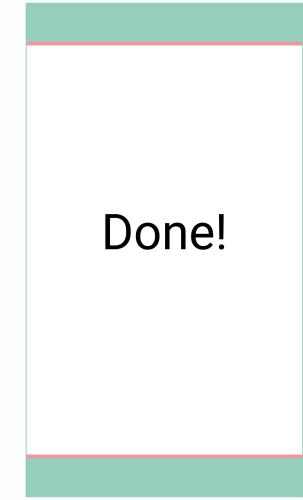
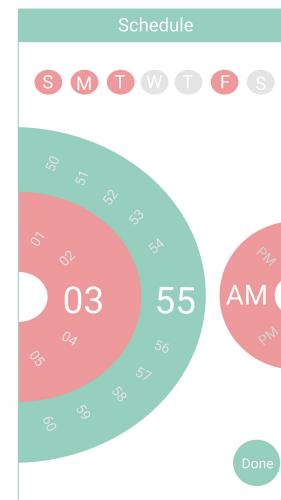
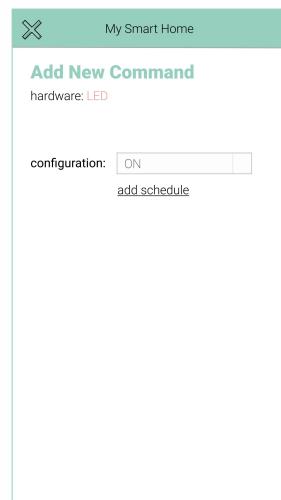
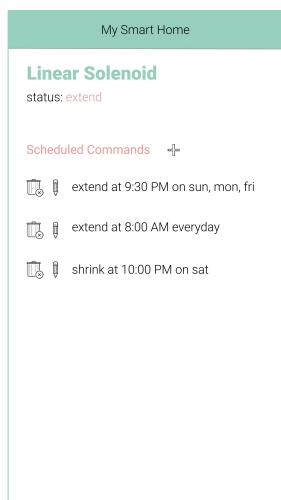
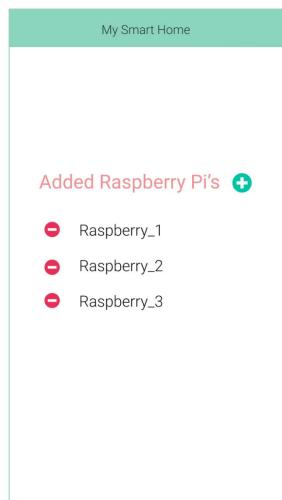
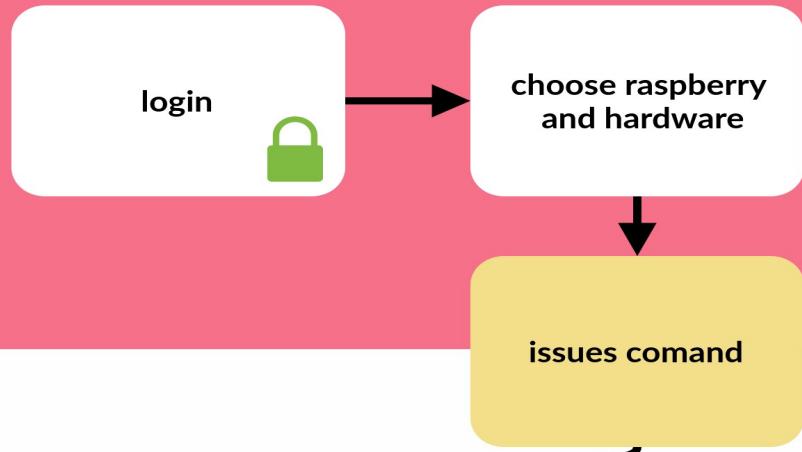


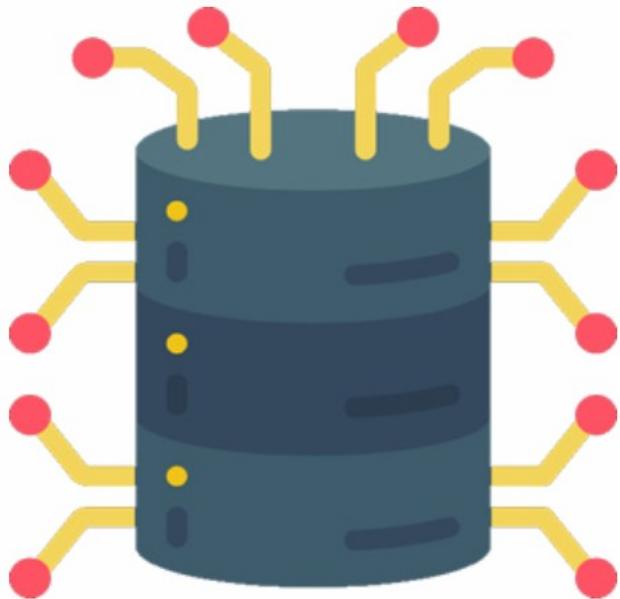
Related Work - Future



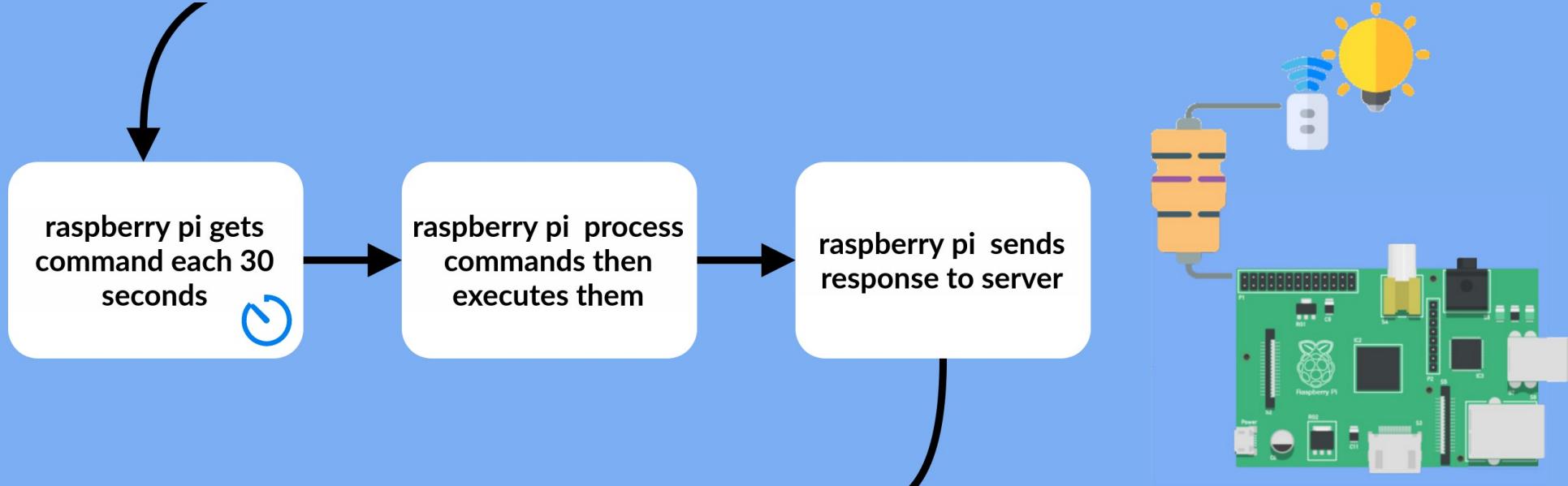
Methodology

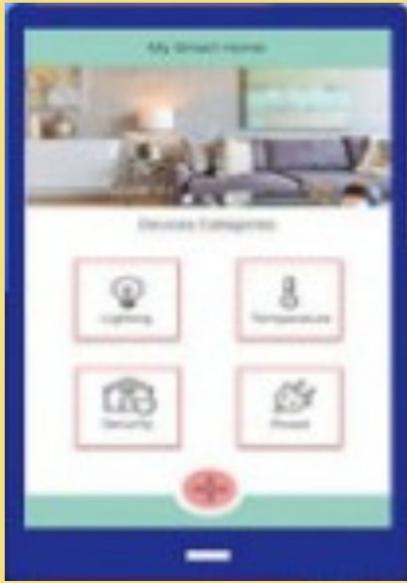






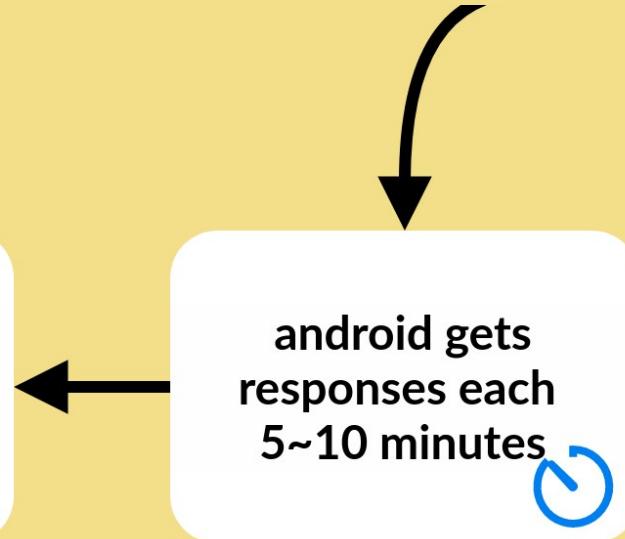
**web app receives
command and save it
in the database**



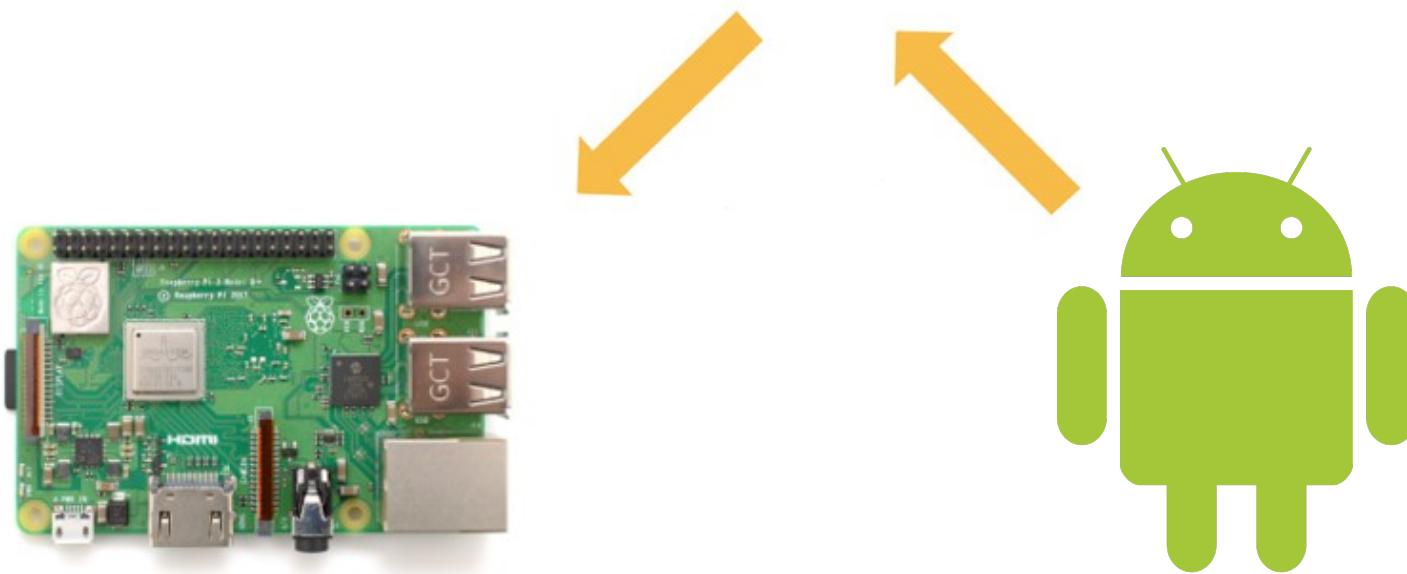


android shows
responses as push
notification

android gets
responses each
5~10 minutes



Implementation





DigitalOcean

```
1 server {
2     listen 80;
3     server_name gp.reem-codes.com;
4
5     location / {
6         include uwsgi_params;
7         uwsgi_pass unix:/home/gin/gp_api/gp.sock;
8     }
9 }
10
```

PROJECTS
Create new record

Gin 1
A AAAA CNAME MX TXT NS SRV CAA

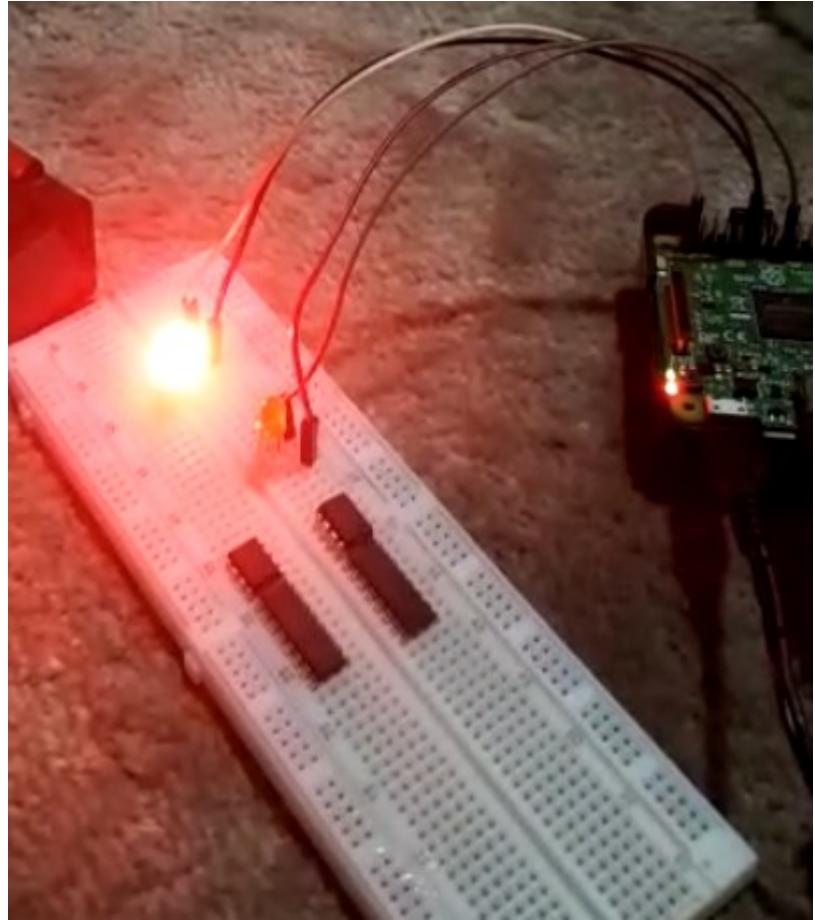
Use @ to create the record at the root of the domain or enter a hostname to create it elsewhere. A records are for IPv4 addresses only and tell a request where your domain should direct to.

HOSTNAME	WILL DIRECT TO	TTL (SECONDS)
<input style="width: 100%; height: 30px; border: 1px solid #ccc; padding: 5px;" type="text" value="Enter @ or hostname"/>	<input style="width: 100%; height: 30px; border: 1px solid #ccc; padding: 5px;" type="text" value="Select resource or enter custom IP"/>	<input style="width: 100px; height: 30px; border: 1px solid #ccc; padding: 5px;" type="text" value="Enter TTL
3600"/> ✓

```
gin@reem: ~/gp_api
File Edit View Search Terminal Help
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
    Use a production WSGI server instead.
* Debug mode: off
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
^C(venv) gin@reem:~/gp_api$ nano wsgi.ini
(venv) gin@reem:~/gp_api$ nano /etc/systemd/system/gp.service
(venv) gin@reem:~/gp_api$ pwdir
No command 'pwdir' found, did you mean:
  Command 'owddir' from package 'ow-shell' (universe)
pwdir: command not found
(venv) gin@reem:~/gp_api$ pwd
/home/gin/gp_api
(venv) gin@reem:~/gp_api$ nano /etc/systemd/system/reem.service
(venv) gin@reem:~/gp_api$ nano /etc/systemd/system/gp.service
(venv) gin@reem:~/gp_api$ 
(venv) gin@reem:~/gp_api$ sudo nano /etc/systemd/system/gp.service
(venv) gin@reem:~/gp_api$ sudo systemctl start gp
(venv) gin@reem:~/gp_api$ sudo systemctl enable gp
Created symlink from /etc/systemd/system/multi-user.target.wants/gp.service to /etc/systemd/system/gp.service.
(venv) gin@reem:~/gp_api$ sudo nano /etc/nginx/sites-available/gp
(venv) gin@reem:~/gp_api$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
```



```
Total 4 (delta 3), reused 0 (delta 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects
To https://github.com/reem-codes/gp_rp.git
  d92cd75..cc6093a master -> master
pi@gin:~/Desktop/gp_rp $ python main.py
6
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpVXQiOjE1ODI1MzQ0
xpQeoPHQxyYu6YTatg47Z33D1RINDH3E6bL6qI38
{"schedule": {"time": "1:45 PM", "updateAt": "Thu, 27
0:47:18 GMT", "id": 37}}
I'm executing..
(11, True)
main.py:17: RuntimeWarning: This channel is already in use
  GPIO.setup(_gpio, GPIO.OUT)
{"message": "resource added successfully", "object": {"u
12:09:40 GMT", "id": 48, "isDone": True, "command_id": 1
{"message": "resource updated successfully", "object": {"id": 14, "raspberry_id": 6, "icon": "light"}}
```

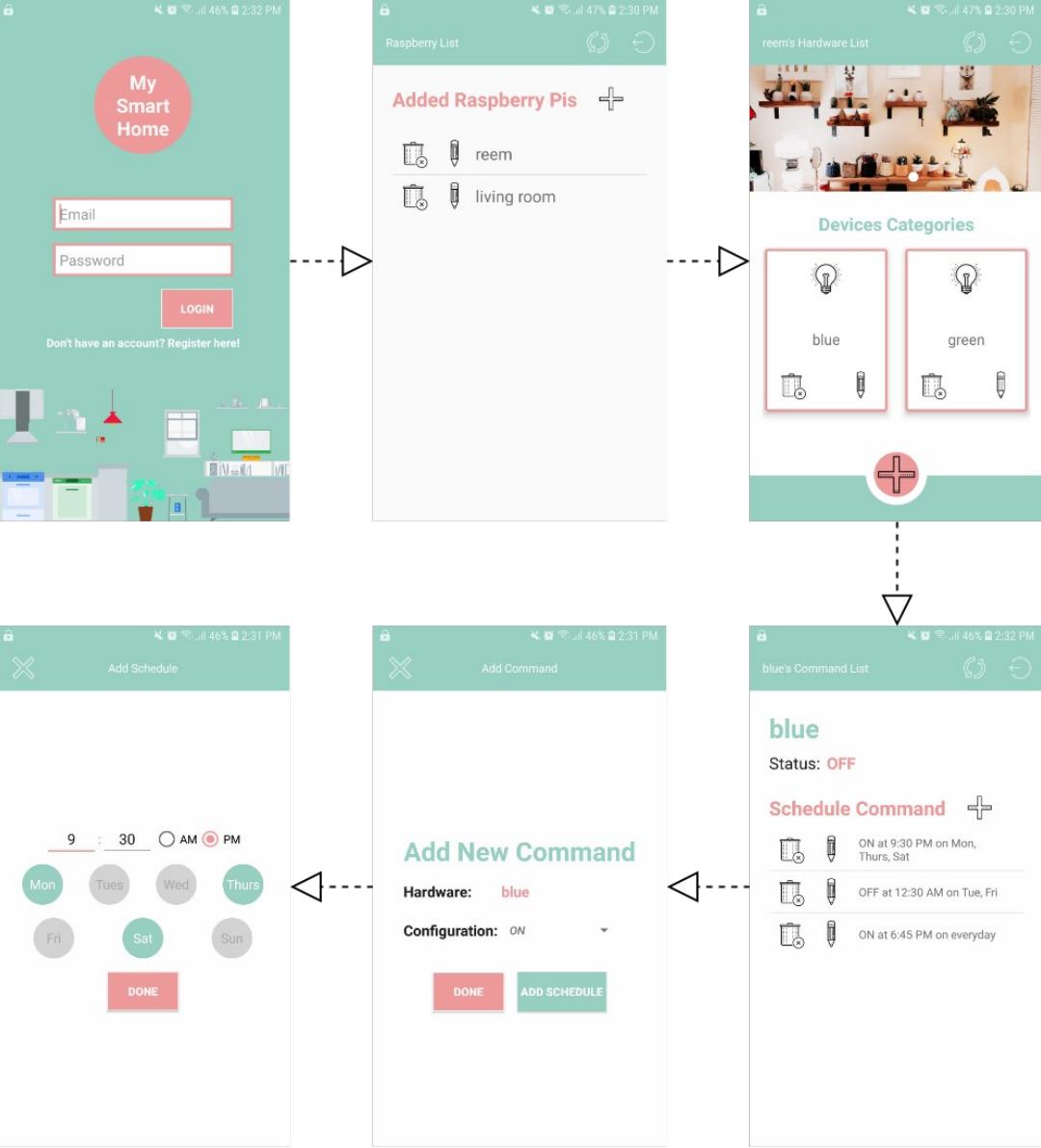




⚙️ My Smart Home • Just now ^

Command executed successfully

command executed successfully
Thu, 27 Feb 2020 15:11:57 GMT



Testing and Validation - Validation

resource updated successfully

resource deleted successfully

Schedule added

You must enter a password

Bad email or password

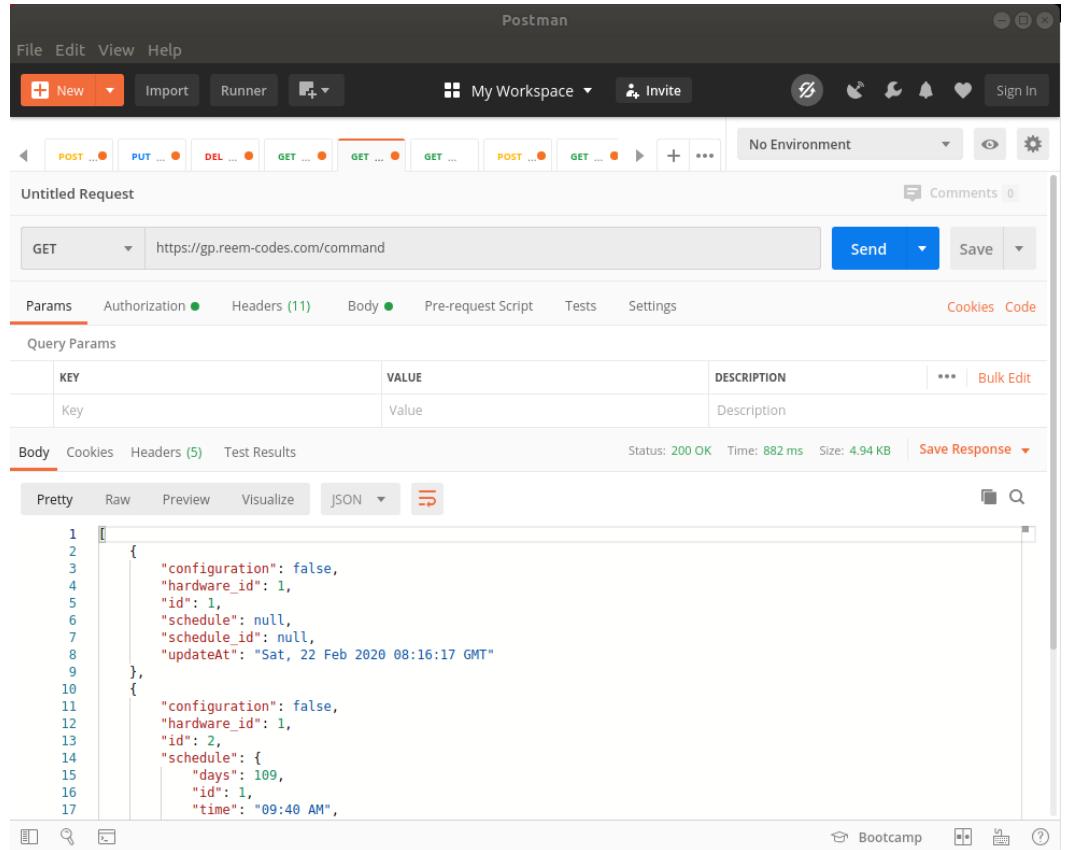
email is incorrect

please check network and try again

user cancelled

Testing and Validation - Testing

No.	Function	Input	Expected Output	Resulting Output
1	Login	email and password	If exists in the database and credentials correct, go to raspberry pi's activity.	Display raspberry pi's activity.
2	Adding a resource	resource attributes	Add a new resource if data is in the right format and required fields are filled.	Return successfully added message
3	Editing a resource	resource id and its attributes	Edit resource if id is correct and data is in the right format and required fields are filled.	Return successfully edited message
4	Deleting a resource	resource id	Delete resource if id is correct.	Return successfully deleted message

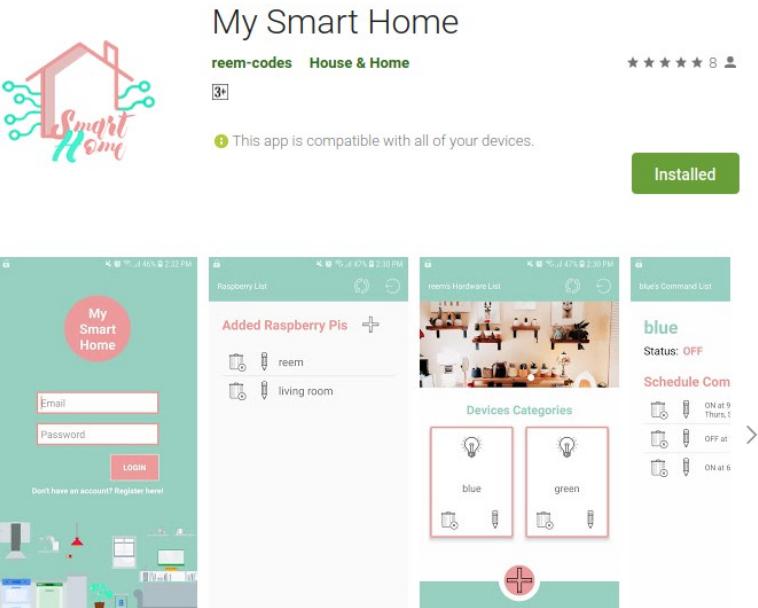


Testing and Validation - Testing

Requirement	Success	Failure	Description
User login or register	✓		validation and authentication are implemented too.
Manage resources	✓		a user can get, delete, create and edit each of the following: raspberry pi, hardware, command and schedule.
Receive Responses	✓		Response service pulls responses from the server each 5 minutes and show them in the notifications.

Requirement	Success	Failure	Description
Get command list	✓		raspberry pi read commands each 30 seconds.
Update local queue	✓		The local queue was replaced by a list for faster implementation
Execute commands saved in queue	✓		Raspberry pi execute commands accordingly
Submit a command response to server	✓		Once done, a response is POST to the server

Deployment - Android



This is my graduation project. It is a system that aims to control any hardware online as long as it is connected to a registered raspberry pi.

ABSTRACT:

With the recent very rapid progress in technology and automation, and towards easier daily life tasks, there has become a need for remote control of almost all possible aspects of living. Some examples

[READ MORE](#)

The image shows the Google Play Console interface. At the top, it says 'All applications'. On the left, there are navigation links: 'All applications', 'Game services', 'Download reports', 'Alerts', and 'Settings'. A message in a box states: 'Due to adjusted work schedules at this time, we are currently experiencing longer than usual review times. Please expect review times of 7 days or longer.' On the right, there is a 'CREATE APPLICATION' button. Below these are two tables. The first table shows a single application entry: 'My Smart Home' by 'com.reem_codes'. It has 7 active installs, a 5.00 Google Play rating, was last updated on Mar 24, 2020, and is in a 'Published' status. The second table is a 'Filter' table with columns: App name, Active installs, Google Play rating, Last update, and Status. The page footer says 'Page 1 of 1'.

https://play.google.com/store/apps/details?id=com.reem_codes.gp_android

My Smart Home
This is my graduation project it's a smart home controller using raspberry pi.

Methodology

- Android Application
- Web Application
- Raspberry Pi

The main application works on the user interface, its goal is to receive requests and send them to the raspberry pi. The raspberry pi will receive the request and execute the command. After executing the command it will send a response back to the user. User can see the status of each device. Also, creating the user interface is very important because it's the first thing that user will see when he wants to use the system.

Gallery

- Smart Home Control Panel
- Add New Hardware
- Schedule Command
- Add New Command
- Hardware Control
- Physical Control

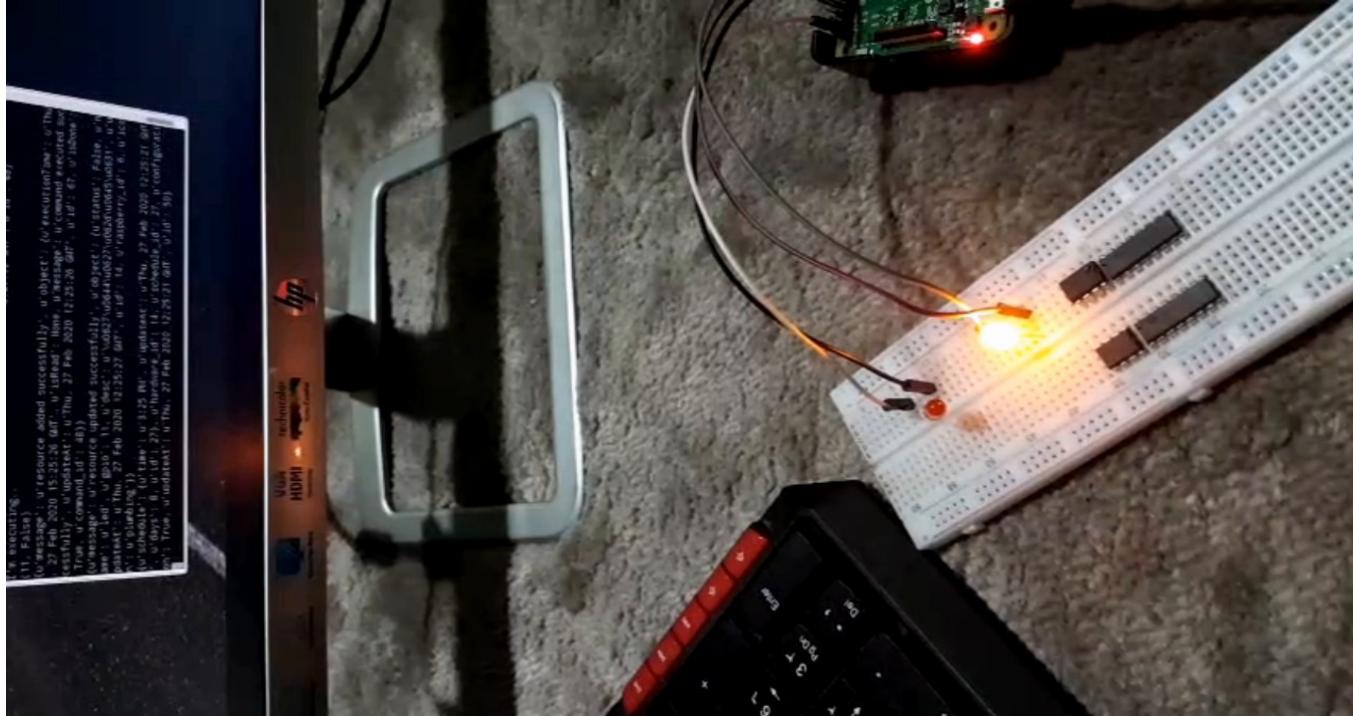
Deliverable

- Scientific Paper
- Slide Show
- Report
- Android Application
- Raspberry Pi Console Application
- Web Application

Deployment – Web Application

<https://gp.reem-codes.com/>

Conclusion - Evaluation



Conclusion – Problems Faced

- Time management: making 3 applications
- Configuring the server
- Publishing application to google play

Conclusion – Future Work

- Add configuration table
- Test on multiple devices
- Make an iOS version

THANK YOU!

- 436200058 Abeer Ahmed Ezzat
- 436200063 Doha Nidal AlZouhbi
- 437004005 Mona Saud AlKhathlan
- 437004100 Nouf Abdullah AlDajani
- 437004875 Reem Ali AlGhamdi
- 436006939 Sarah Khalid AlHussain

