autoHome Demo #1 Contributions

Group #1

Calvin Chiu

Elie Rosen

Elvison Dominguez

Paul Kania

Rohith Dronadula
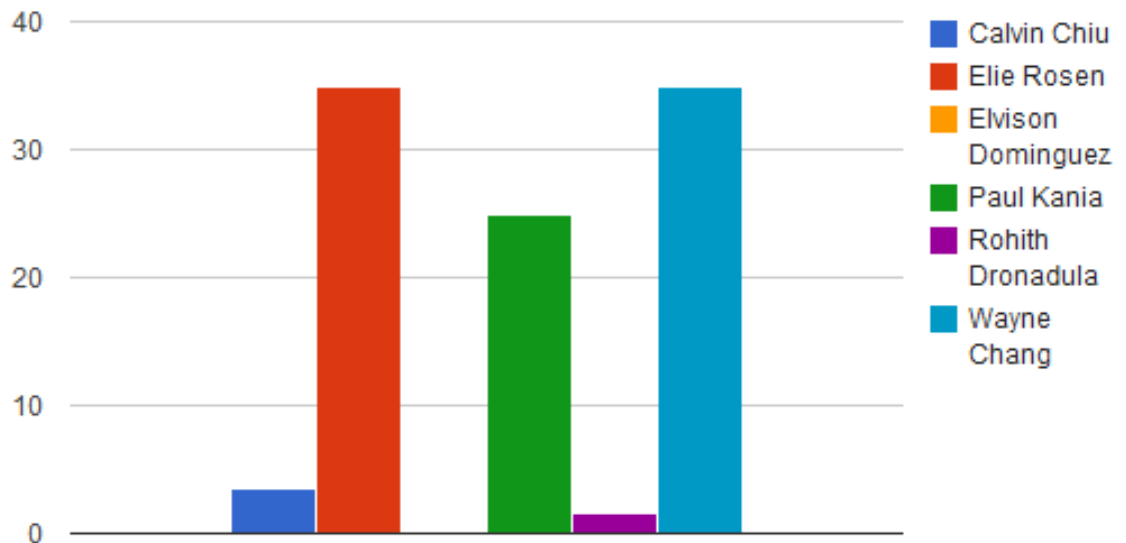
Wayne Chang

http://autohome.mylifeiscomputers.net:3000/dashboard
https://github.com/autohomeproject

Responsibility Matrix

Responsibilities	Calvin Chiu	Elie Rosen	Elvison Dominguez	Paul Kania	Rohith Dronadula	Wayne Chang	Totals	Possible Points
CODING	5%	45%			5%	45%	100%	30
UNIT TESTING		50%				50%	100%	10
WEB DESIGN		100%					100%	5
INTEGRATION		50%				50%	100%	10
DEBUGGING		50%				50%	100%	5
DOC'S		50%		25%		25%	100%	10
DATA COLLECTION	25%	25%		25%		25%	100%	8
DATABASE						100%	100%	5
BROCHURE			5%	95%			100%	2
SLIDES				100%			100%	3
PROJECT MANEGMENT		33%		34%		33%	100%	12
Totals:	3.5	35	0.1	25	1.5	35	100%	100

Program code writing
This web based software component of autoHome has been made entirely using Ruby on Rails and a PostgreSQL database. Elie and Wayne designed and implemented most of the site including the login page, dashboard, devices, device types, rooms and locations, configure rules and users pages along with the edit/show/destroy and add pages that accompany all of the tables. Rohith designed and implemented the logs and alerts page. Calvin designed and implemented the Backup / Restore page.

All code updates can be seen in log form here:
https://github.com/autohomeproject/code/commits/master

In the following demo, we will implement the hardware based component of our project in which the website will be able to interact directly with physical components

Unit testing
Code for testing was automatically generated by the Rails programming as a result all of the tests for each page was automatically generated from the models and performs all of the use tests. This was done by Wayne.

The UI was linked to the models and then tested manually by trying all of the features in every conceivable way possible this made sure that every page displayed and worked properly. This was done by Elie.

Webpage design
All of the styling of the website was made by Elie. Through use of designing a template it allowed for rules to be made on how to design and follow for other pages, this is similar in practice to how Applications for Apple's iOS forces apps to be consistent.

Integration and integration testing, cross-platform issues
The integration that this project consisted of included Wayne working on a lot of the backend implementations, while Elie joined the UI to those backend processes, thus creating a final project which worked well and looked good.

The website runs on a Rails server it is very easy to set up Rails on Windows / Linux / and Mac OS. The site can handle being run on tablets such as the iPad and Android tablets through a web browser.

Testing algorithms
The testing script is made by rails after generating all of the files, this ensure that no memory leaks occur and that the database runs properly. This was done by Elie and Wayne.

Debugging

Throughout the course of the programming session many small errors appeared through use of unfamiliar syntax since Ruby and Rails was new to us. When Rails compiles it spits out detailed error messages which can then be used to debug the code. All errors were fixed when a unit test failed. This was performed by Elie and Wayne.

Program documentation,
README1, README2, README3, README4 and Individual_Contributions were created by Elie

Technical_Documentation was done by Wayne

User_Documentation and Brochure description was done by Paul

Designing database tables and maintaining the database
Wayne designed the database and implemented it through PostgreSQL, the database worked really well with minimal debugging and came together really well especially while using technology that we have never worked with before.

Data collection
Data for the site came from inputting values into the database tables; this was performed by Calvin, Elie, Paul and Wayne. The data seen across the website in the tables is the collected data created by us.

Brochure/flyer preparation
Paul created the brochure and wrote all of the information from scratch, proofread it, and designed the template.

Slides
Paul created the slides to easily work with the flow of our demo in which the brochure was first presented then the slides which then allowed us to fluidly slide into showing off the product.

Project management
Elie, Paul, and Wayne worked to make sure that the demo was finished in time.

Elie created the github repository, and sent out update emails and installed Rails
Paul sent out update emails and emails to designate tasks
Wayne set up PostgreSQL and worked on the github repository and sent out update emails