

Declaration Cover Sheet for Project Submission

SECTION 1 Student to complete

Name: Alan Leonard
Student ID: 15012492
Supervisor: Eugene McLaughlin

SECTION 2 Confirmation of Authorship

The acceptance of your work is subject to your signature on the following declaration:

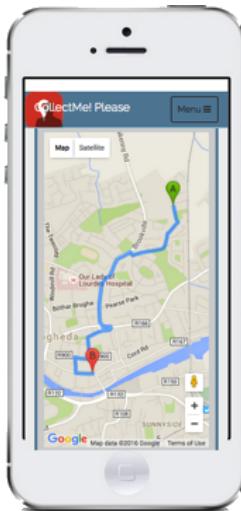
I confirm that I have read the College statement on plagiarism (summarized overleaf and printed in full in the Student Handbook) and that the work I have submitted for assessment is entirely my own work.

Signature:_____

Date: 03/08/2016

NB. If it is suspected that your assignment contains the work of others falsely represented as your own, it will be referred to the College's Disciplinary Committee. Should the Committee be satisfied that plagiarism has occurred this is likely to lead to your failing the module and possibly to your being suspended or expelled from college.

Complete the sections above and attach it to the front of one of the copies of your assignment,



CollectME!

Application Report

Higher Diploma in Science in Web
Technologies

Includes:

- Testing documentation
- Requirement Specifications documentation
- Analysis and design documentation

Alan Leoanrd
03/08/2016

Table of Contents

Introduction	3
Why	4
What	4
Who	6
Register and Login	7
Google Map and Geo Location	7
Chat	8
Home, Contact, Register, login pages	8
Testing	8
Requirements Specifications	22
Purpose	22
Project Scope	22
User Requirements Definition	22
Requirements Specification	23
Functional requirements	24
Use Case Diagram	24
Requirement 1 <Collect me button>	24
Requirement 2 < Login and get coordinates >	27
Non-Functional Requirements	28
Performance/Response time requirement	29
Security requirement	29
Interface requirements	29
GUI (mock-up or prototype)	29
System Architecture	31
System Evolution	32
Analysis and design documentation	33
Purpose of The Application Report Document	33
General Overview and Design Guidelines/Approach	33
Assumptions / Constraints / Standards	33

Logical View	35
Software Architecture	35
Use-Cases – part II	36
Database Design	36
Data Conversions	37
Application Program Interfaces	37
Child desktop	37
Child Mobile view	37
User Interface Design	38
Performance	39
Appendix B: Key Terms	39
References	40

Introduction

The idea for this application came in many different forms. The original idea was just to have a geo location icon show the position of the child or young teen. As the app developed especially through the testing phase other functionality was added to improve the overall apps performance. Items like chat, get flight path (disabled now), google places, google Directions and Google's direction text. Through each segment or iteration of the system life cycle the application grew in directions that I was not expecting it to go. I personally got to learn a lot from this project like: 'Geo location API call backs.' and angular JS dependency injection were a few of the main learning curves. The in class function specifications documentation and system requirements were very helpful. They made this application go in the right direction and gave a sense of professionalism to the overall project.

Why

I feel the methodology for this application is a need for a helpful tracking device especially for young teens and children who are out and about with friends while at playgrounds, fairs, shopping malls, fun attractions, carnivals or just coming home from school. Parents worry about their children and while not being too invasive this application is helpful for both the parent and child. The app can act as a taxi service for the child while giving the parent a detailed position for the child's exact location. The application has a chat facility that will enable both parent and child to interact with each other. Through the text feature instant messages can be sent and the user can also see their parents or child's position on the map provided. There is also a 'nearby' facility that can find the nearest ATM, School, Garda station, Cinema, Stores and many more (safe) public locations that the parent can send their child to be picked up.

When ready to be collected the child simply presses a 'collect me' button and the parent will get an alert on their screen to tell them that the child is ready to be picked up. The parent simply looks for the child's current location and they can chat to arrange a place to be collected. An email will be sent as soon as the child presses the 'collect me button' notifying their guardians of their intent to be collected.

Some sections of society love to live out their lives on Facebook or Twitter, They like to display their lives as exciting, different or even unique. This unfortunately can lead to people being watched without them even knowing or becoming friends with someone who they don't really know or think are the same age, race and gender. To most children this is harmless but there is a danger with this and it is better to monitor online behaviour at all times if possible. The app can give the parents some knowledge of their child's whereabouts and leave them with a sense of knowing.

What

The application is a web based single page application driven by the angular framework that is behind popular sites like YouTube on PS3 and Gmail, a good way of knowing if it is an angular site just look at the address bar for a trailing '#' sign. The app is set using an MVC or Model,

view and controller framework. The Model is MySQL database and tables, the view is angular html ng-directives, html pages, partials and the controller is glue that holds it all together. The controllers are set up in this application in a file called alan.js. This file is the brains behind the whole application. The file itself has a couple of controllers that do very different tasks from managing the registration, validating the contact page and sending emails, but the main controller will handle the login, maps display and chat functionality. Sample directives include ‘ng-app’ for initializing the application, ‘ng-model’ for creating a variable on the scope, ‘ng-repeat’ for looping through data, ‘ng-show’ and ‘ng-hide’ to manipulate the DOM.

Angular uses a {{}} or ‘moustache’ syntax to display variables or data created within controllers. This is very user friendly and can allow the user to create apps quickly with angular’s popular data-binding ability. Another great feature within angular is the filter functionality, filters limit data, separate content types and add currencies or dates to related data. To use angular I simply added the relevant CDN to the top of my index page. Angular has different CDN’s for animation, location, routers and timeouts, some of which I used within collect me.

Different views will be displayed depending on what is typed in the URL. This is due to the angular router. A ‘Router’ dependency injected into a controller to provide routing and redirecting throughout the ‘site. I used other dependencies like the ‘\$http’, to contact API’s or Json files that I have data in. The \$timeout allowed me to set intervals between functions, this gave me updated information at whatever intervals the user wants, and example of intervals are 7 minutes, 15 minutes, one hour and one day.

PHP / MySQL is use to connect the application to the database. I have various PHP scripts within the partials and includes folders that with MySQL connect to certain tables and retrieve the relevant information. There are PHP scripts for getting ‘chat’, getting ‘customers’, inserting ‘customers’, getting ‘positions’, sending ‘emails’, in fact most of the CRUD operations are using PHP/MySQL scripts.

I setup an online server ‘<http://tigrimigri.com/>’ that Is hosting the applications website ‘collectme.tigrimigri.com’ and is also hosting phpMyAdmin software that allows me quick access to MySQL databases and tables. The database is simple made up of 3 tables “customers”, “positions” and “chat” that when work together create a dynamic and informative application.

Collect me applications is set to work on most browsers (*except chrome - due to depreciation of functionality in 2016 see footnote¹*) all tables and all mobile phones. HTML5, CSS3 are the backbone of this project and allow the application to become fully responsive. Bootstrap is used but most if not all of its CSS selectors have been modified or changed to enable styling for the collect me style palette.

Who

When starting to build my project I had to gather the correct requirements (see attached) and see what functional specifications (see attached) would have to be implemented to get the application working correctly. I began creating the application with angular. I laid out a site structure with an index.html page and the rest of the pages are includes or partials. The style, script and images folders were created as if it were a standard website. The structure of main pages are as follows:

Site Structure

Index.html

Partials >>

- Mainpage.html
- Contact.html
- About.html
- Login.html
- Registar.html
- Admin.html

Includes >>

- getChildsLocation.php
- getcontactemail.php
- getcustomers.php

CSS >>

¹ <https://developers.google.com/web/updates/2016/04/geolocation-on-secure-contexts-only?hl=en>

Bootstrap.min.css
Style.css
JS >>
Alan.js
Images >>

Although there are more pages in the site structure I have just included the main ones to give you an idea of the layouts involved. The next step was to create some MySQL through the software called phpMyAdmin. I created the tables with primary keys and the suitable data types. To connect the application to the tables I used mysqli_connect() functions and was able to retrieve any information necessary. Following on in the build process was to include the google map functionality. Firstly I created the register and login functionality

Register and Login

I created a login and registration pages that connect to the database. If you log in as a parent you are shown a different map to the one shown to the child. This is done to keep the two locations totally separate when getting directions from the google API service.

Google Map and Geo Location

W3Schools provides Geo location as its introduction to HTML5, so it is readily available to everyone. Although as mentioned earlier for security reason the functionality has been depreciated in the Google chrome browser. Even though it does not work correctly in this browser it works fine in Firefox, Safari, Opera, Windows Explorer, all tables and mobile phones (see test section for results).

When Geo locations would collect the person's coordinates through the relevant browser the next step was to update the database tables with the user's exact location.

I used Angular's http request to get back the locations and display them on their dedicated maps. Most of the code is provided by w3schools and is pretty simple to include. The only difference with my application is my variables are \$scope variables when using angular. Once I had the locations and the maps it was now time to utilize any and all Google API's available. I

used flight Path (not in use now), Google Places for nearby places of interest and Google Directions to show directions both as an image on the map and in step by step text form.

Chat

The chat function was set up by sending and receiving users posts to and from the chat table in the database. A set timeout functions is working in the background timing when and how long the intervals are set between when the functions fire off. The chats GUI will need more work after the project is finished as this is a time consuming item but will need to be done to make the GUI look right. I have it working and you can see each post as it arrives, but as I say – a little more time spent on it and it would be perfect.

Home, Contact, Register, login pages

As this application is a Single Page Application the index page is very small and all the partials or includes are displayed in a div called “view”. The home, contact, login, register and admin pages are not static. The homepage or ‘mainpage’ as it is called in the partials folder gets information from a json file and uses angular ng-repeat to display the information. The contact page has an angular validated contact form that sends an email to notify the owner of the site if anyone has posted a query. The register page allows a user to sign up and the login page allows the user to login. Node.js, Grunt.js and Bower.js have not been used on this project but can easily be integrated with little or no disruption the overall layout of the site structure.

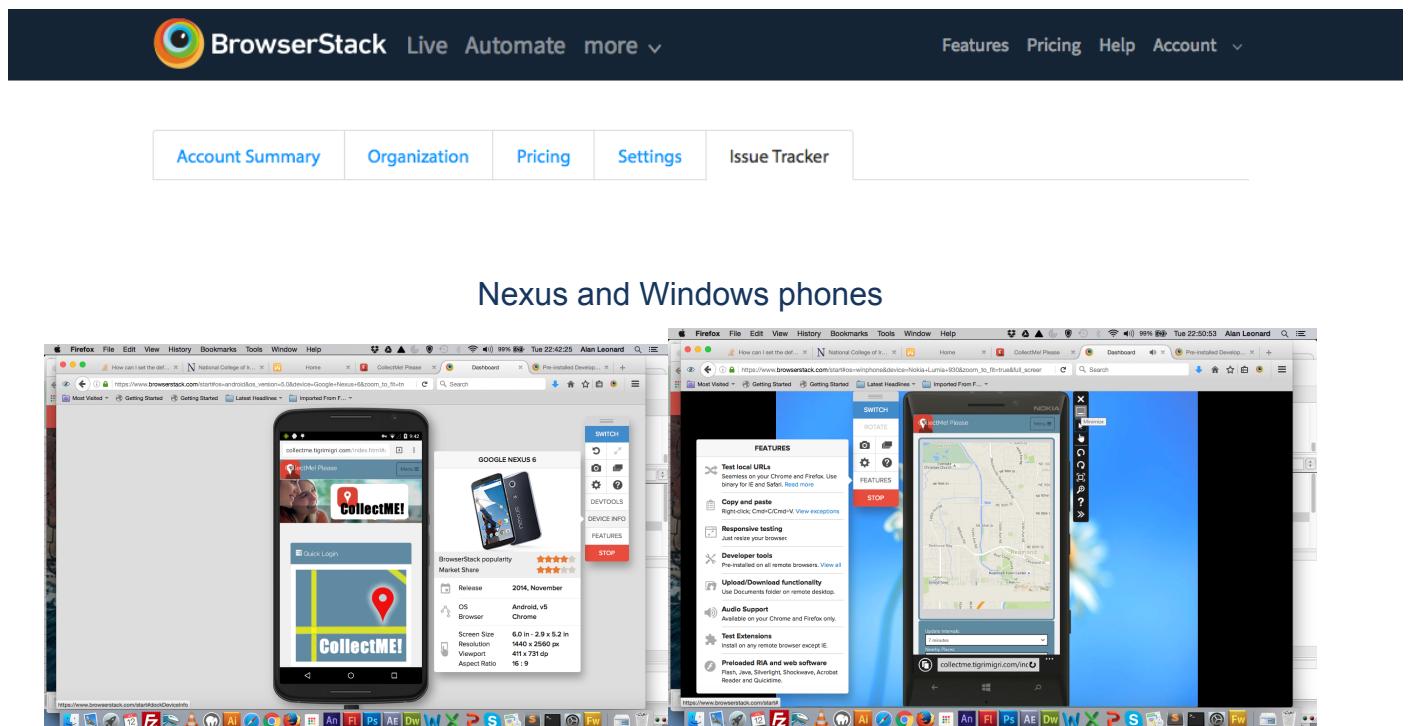
New included features

Some new features have been including as a direct result of testing. I have added a street view of current location and destination location. This will give both parent and child a visual on the location where they are to meet.

Testing

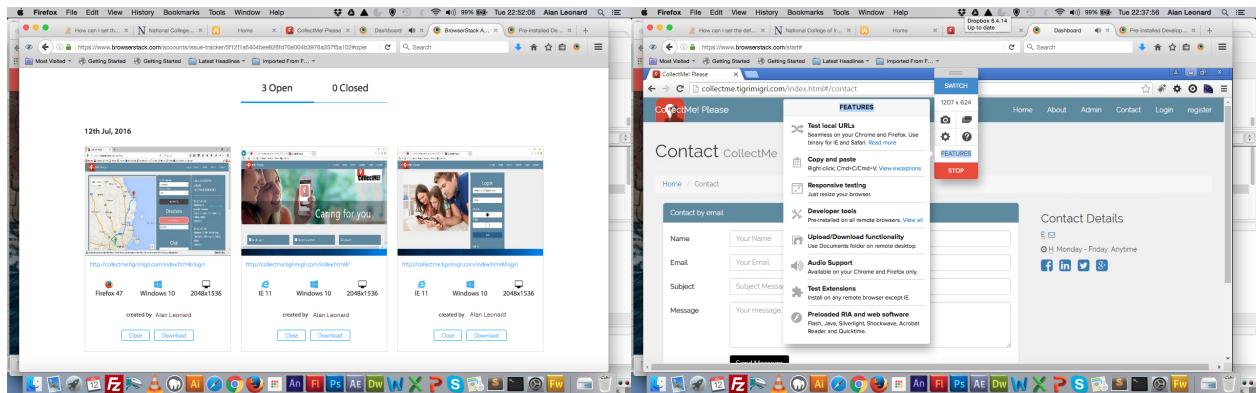
Testing has been a big part of this project. Without testing there would be no application. The app uses Geo location and set timeout functions to gather information, this needs to be tested on the move for best results. As I live in Drogheda I use the train every day to get to work/college. Using the train for my Geo location tests were perfect. From the offset I had problems with the set timeouts, collect me functions and even the styling for the mobile view. Testing was done daily and after each test I found it helpful for the next iteration state of the build. I found that the application did not work in all browsers so I set out to test them all. The following screenshots and results show the indebt testing the application went through to get to its present stage.

Browser testing – <https://www.browserstack.com/>



Chrome and Windows FireFox

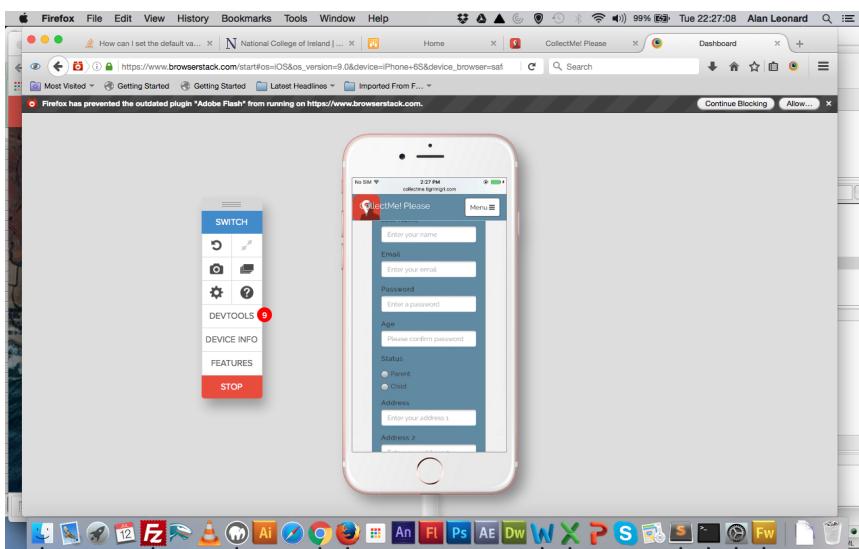
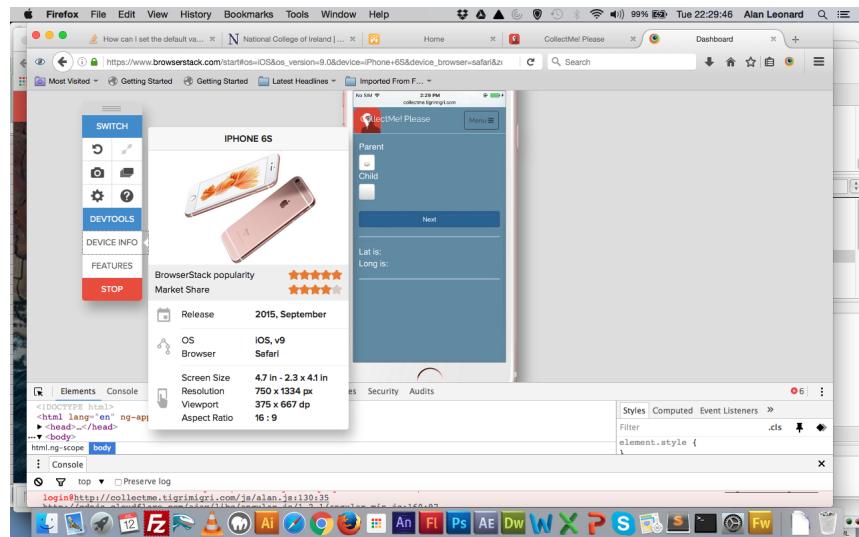
Requirements Specification



Postman CRUD application and Firefox

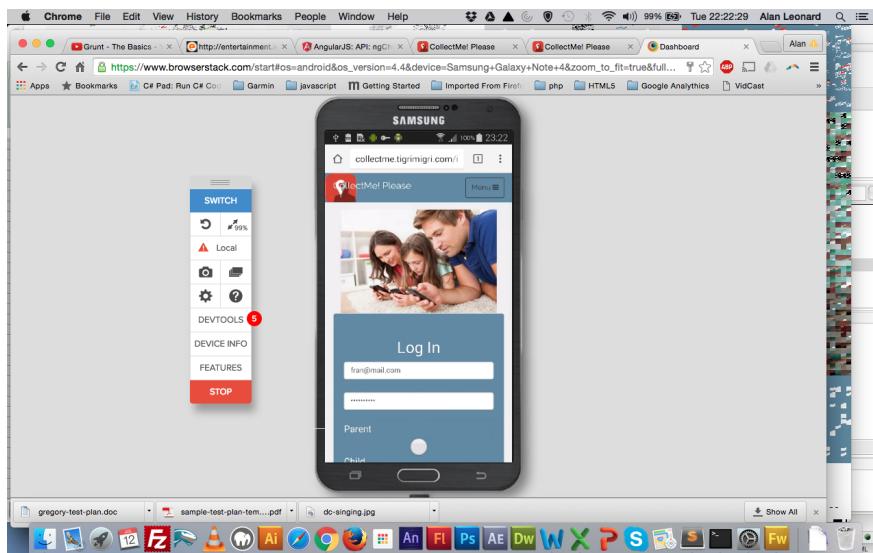
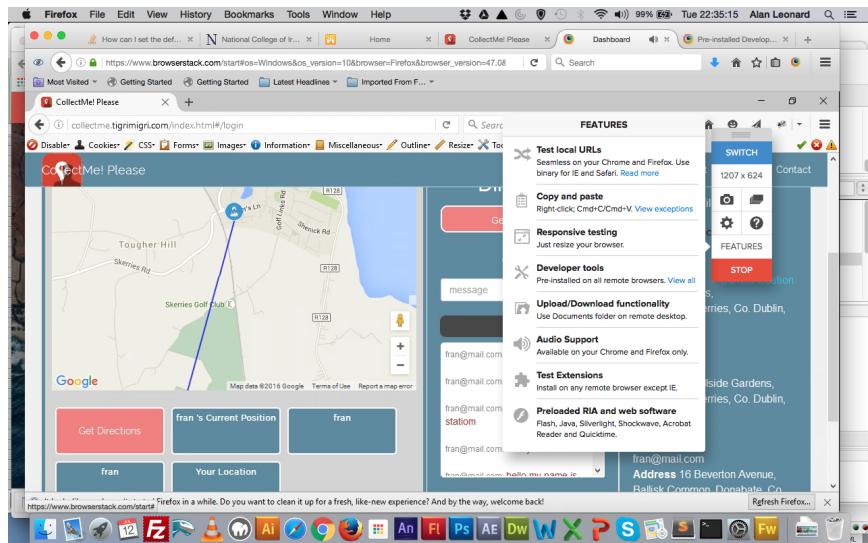
iPhone6

Requirements Specification



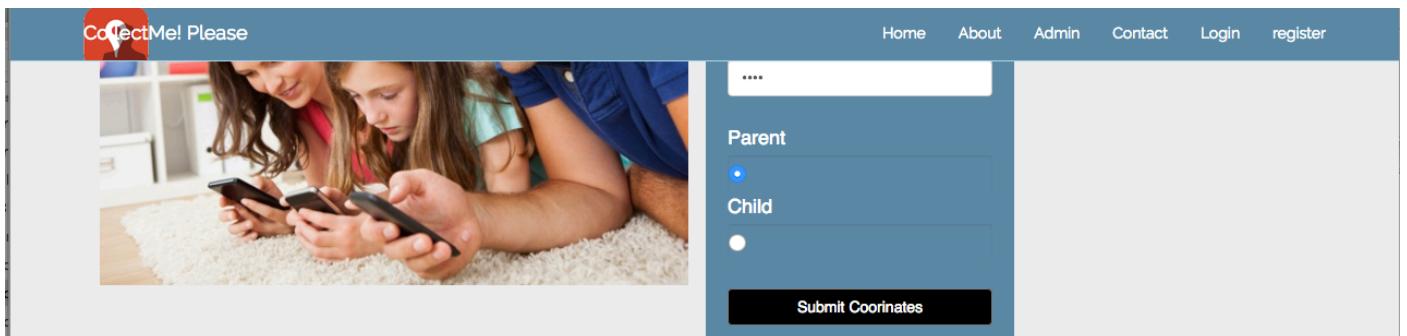
Android and firefox

Requirements Specification



Chrome geo location fail

Requirements Specification

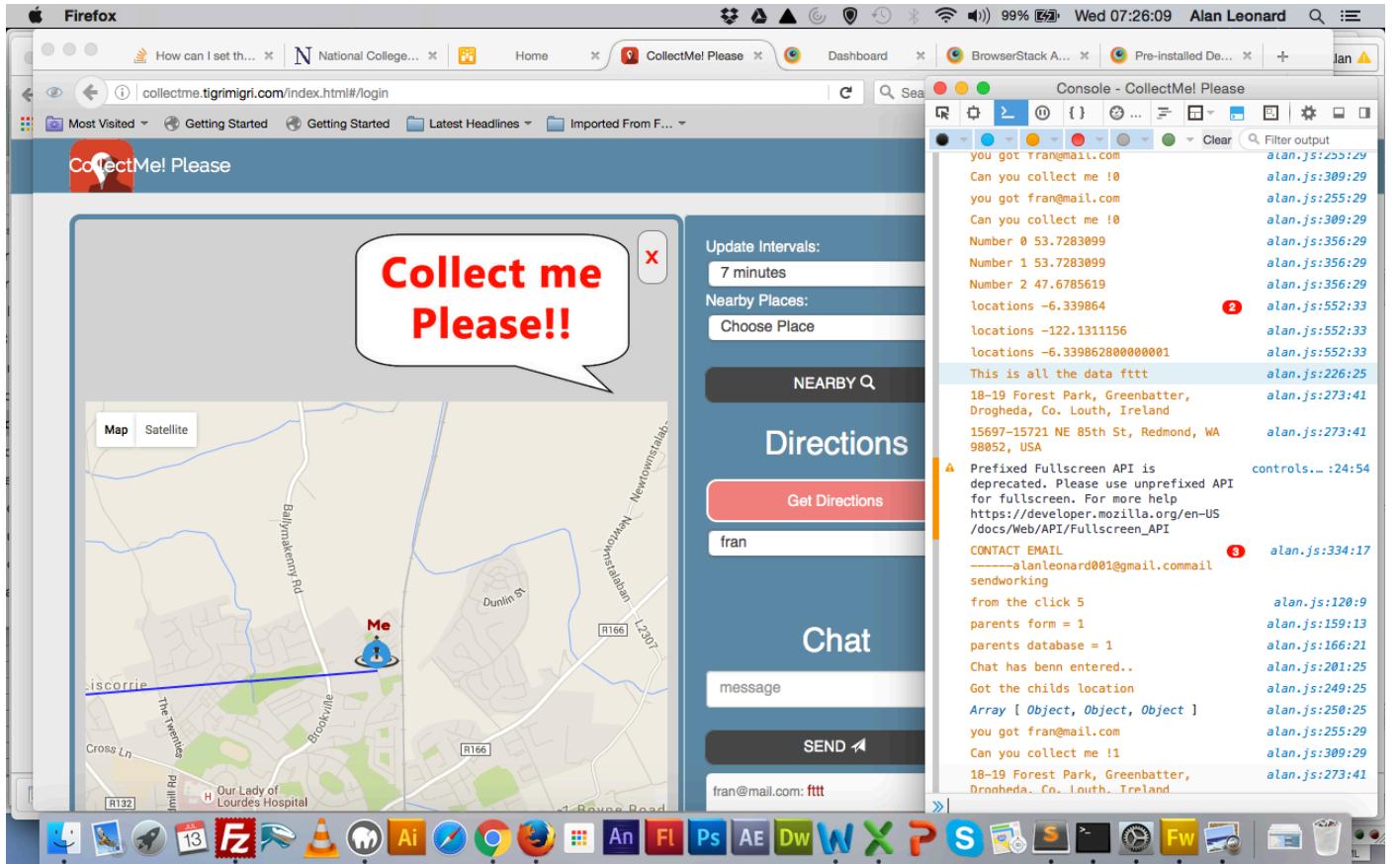


The screenshot shows a web application interface. At the top, there's a navigation bar with links for Home, About, Admin, Contact, Login, and register. On the left, there's a logo for 'CollectMe! Please' featuring a red icon of a person. The main content area has a large image of a family (two adults and two children) looking at smartphones. To the right of the image is a sidebar with the word 'Parent' and two radio buttons labeled 'Parent' and 'Child'. Below this is a 'Submit Coordinates' button. At the bottom of the page is a browser's developer tools console window. It shows a warning about deprecated methods and several error messages, including a TypeError from angular.js:9101.

```
⚠ getCurrentPosition() and watchPosition() are deprecated on insecure origins. To use this feature, you should consider switching your application to a secure origin, such as HTTPS. See https://goo.gl/rStTgZ for more details.  
from the click undefined  
angular.js:9101  
TypeError: Cannot read property 'coords' of undefined  
at g.$scope.login (http://collectme.tigrimigri.com/is/alan.js:130:35)  
at http://cdnjs.cloudflare.com/ajax/libs/angular.js/1.2.1/angular.min.js:160:92  
at http://cdnjs.cloudflare.com/ajax/libs/angular.js/1.2.1/angular.min.js:177:83  
at g.$eval (http://cdnjs.cloudflare.com/ajax/libs/angular.js/1.2.1/angular.min.js:100:328)  
at g.$apply (http://cdnjs.cloudflare.com/ajax/libs/angular.js/1.2.1/angular.min.js:101:105)  
at HTMLFormElement.<anonymous> (http://cdnjs.cloudflare.com/ajax/libs/angular.js/1.2.1/angular.min.js:177:65)  
at http://cdnjs.cloudflare.com/ajax/libs/angular.js/1.2.1/angular.min.js:27:15  
at Array.forEach (native)  
at p (http://cdnjs.cloudflare.com/ajax/libs/angular.js/1.2.1/angular.min.js:7:255)  
at HTMLFormElement.c (http://cdnjs.cloudflare.com/ajax/libs/angular.js/1.2.1/angular.min.js:26:492)
```

FireFox console

Requirements Specification



Testing checklist

	A	B	C	D	E	F
1	http://collectme.tigrimigri.com/index.html#/login				Tested by Alan Leonard	
2	Collect me unit testing checklist					
3	Title	Poor	Good	Very Good	Excellent	Date tested
4	Responsive			✓		14 July 2016
5	Internet connection		✓			14 July 2026
6	GUI			✓		14 July 2036
7	Navigation				✓	14 July 2046
8	Load time		✓			14 July 2056
9	Browser firefox				✓	14 July 2066
10	Browser Chrome	✓				14 July 2076
11	Borwser Safari				✓	14 July 2086
12	Desktop(Windows 8)				✓	14 July 2096
13	Lapotop (mac book pro)				✓	14 July 2106
14	Tablet (samsung G7)			✓		14 July 2116
15	Mobile phone (Huawei)				✓	15 July 2126
16	Wifi (own)			✓		15 July 2136
17	Wifi (Train)			✓		16 July 2136
18	Proformance			✓		16 July 2136
19	usability			✓		16 July 2136
20						



<http://www.webpagetest.org>

http://www.webpagetest.org/performance_optimization.php?test=160713_A9_18NG&run=1&ached=0

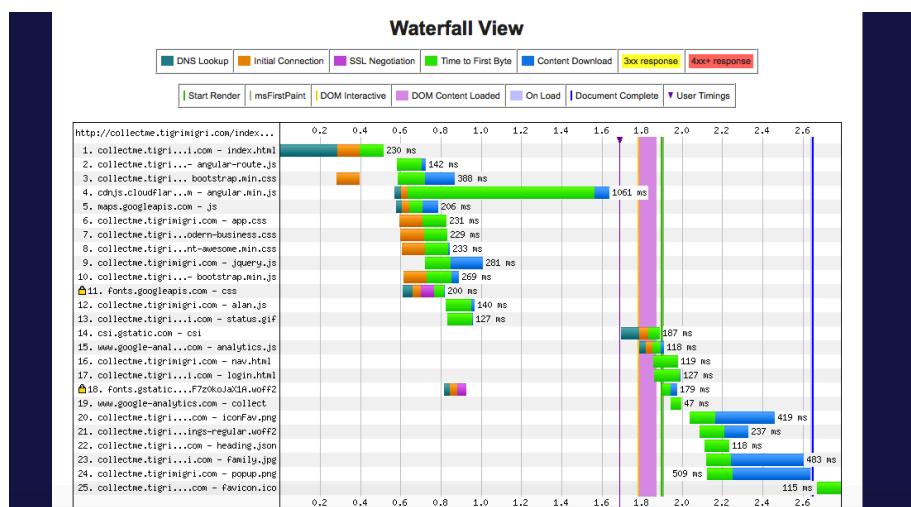
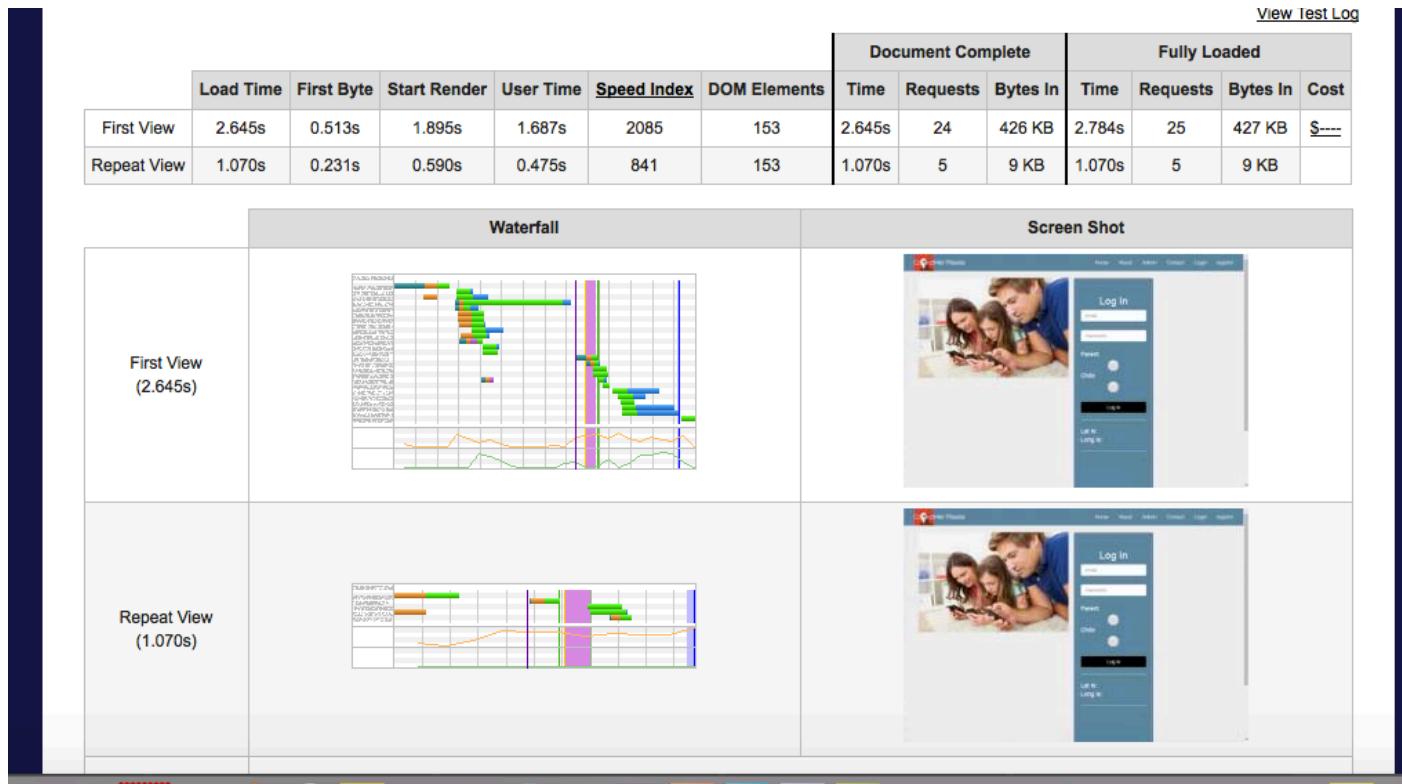
http://www.webpagetest.org/result/160713_A9_18NG/1/details/

Requirements Specification

http://www.webpagetest.org/result/160713_A9_18NG/

http://www.webpagetest.org/breakdown.php?test=160713_A9_18NG&run=1&cached=0

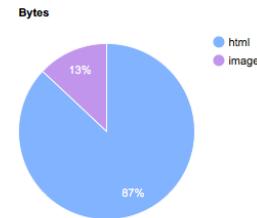
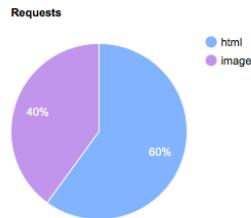
http://www.webpagetest.org/domains.php?test=160713_A9_18NG&run=1&cached=0



Requirements Specification

#	Resource	Content Type	Request Start	DNS Lookup	Initial Connection	SSL Negotiation	Time to First Byte	Content Download	Bytes Downloaded	Error/Status Code	IP
1	http://collectme.tiq...migr.com/index.html	text/html	0.391 s	282 ms	108 ms	-	122 ms	-	1.3 KB	200	94.23.147.18
2	http://collectme.tiq...js/angular-route.js	application/javascript	0.58 s	-	-	-	120 ms	22 ms	8.0 KB	200	94.23.147.18
3	http://collectme.tiq...ss/bootstrap.min.css	text/css	0.587 s	-	110 ms	-	132 ms	146 ms	20.7 KB	200	94.23.147.18
4	http://cdns.cloudfl...1.2.1/angular.min.js	application/javascript	0.631 s	26 ms	33 ms	-	930 ms	72 ms	36.3 KB	200	198.41.215.68
5	http://maps.googleapis.com/css/language=en	text/javascript	0.641 s	26 ms	39 ms	-	64 ms	77 ms	23.5 KB	200	74.125.29.95
6	http://collectme.tiq...modern-business.css	text/css	0.707 s	-	113 ms	-	118 ms	-	1.5 KB	200	94.23.147.18
7	http://collectme.tiq...modern-business.css	text/css	0.712 s	-	112 ms	-	117 ms	-	1.0 KB	200	94.23.147.18
8	http://collectme.tiq...font-awesome.min.css	text/css	0.716 s	-	110 ms	-	117 ms	6 ms	5.2 KB	200	94.23.147.18
9	http://collectme.tiq...gri.com/js/jquery.js	application/javascript	0.723 s	-	-	-	122 ms	159 ms	32.8 KB	200	94.23.147.18
10	http://collectme.tiq...js/bootstrap.min.js	application/javascript	0.727 s	-	111 ms	-	123 ms	35 ms	9.7 KB	200	94.23.147.18
11	https://fonts.googleapis.com/css?family=Rowdy	text/css	0.767 s	44 ms	41 ms	66 ms	46 ms	3 ms	0.7 KB	200	173.194.205.95
12	http://collectme.tiq...migr.com/js/alain.js	application/javascript	0.826 s	-	-	-	122 ms	18 ms	7.1 KB	200	94.23.147.18
13	http://collectme.tiq...om/images/status.gif	image/gif	0.831 s	-	-	-	119 ms	8 ms	3.1 KB	200	94.23.147.18
14	http://csi.gstatic.c...0_10_2_0_8rt=main.17	image/gif	1.826 s	84 ms	47 ms	-	56 ms	-	0.3 KB	204	172.217.1.131
15	http://www.google-analytics.com/analytics.js	text/javascript	1.848 s	27 ms	34 ms	-	41 ms	16 ms	11.6 KB	200	216.58.217.142
16	http://collectme.tiq...om/partials/nav.html	text/html	1.855 s	-	-	-	119 ms	-	0.9 KB	200	94.23.147.18
17	http://collectme.tiq.../partials/login.html	text/html	1.86 s	-	-	-	123 ms	4 ms	2.4 KB	200	94.23.147.18
18	https://fonts.gstatic.com/1.woff2	font/woff2	1.9 s	28 ms	35 ms	44 ms	38 ms	34 ms	20.1 KB	200	216.58.217.131
19	http://www.google-analytics.com/1.js?r=1&z=1784910135	image/gif	1.944 s	-	-	-	47 ms	-	0.4 KB	200	216.58.217.142
20	http://collectme.tiq...s/assets/iconFav.png	image/png	2.039 s	-	-	-	121 ms	298 ms	55.1 KB	200	94.23.147.18
21	http://collectme.tiq...om/fonts/regular.woff2	text/plain	2.089 s	-	-	-	119 ms	118 ms	17.9 KB	200	94.23.147.18
22	http://collectme.tiq...om/json/heading.json	application/json	2.113 s	-	-	-	118 ms	-	0.8 KB	200	94.23.147.18
23	http://collectme.tiq...om/images/family.jpg	image/jpeg	2.119 s	-	-	-	119 ms	364 ms	75.4 KB	200	94.23.147.18
24	http://collectme.tiq...om/assets/popup.png	image/png	2.126 s	-	-	-	120 ms	389 ms	80.8 KB	200	94.23.147.18
25	http://collectme.tiq...gri.com/favicon.ico	image/x-icon	2.669 s	-	-	-	115 ms	-	0.7 KB	200	94.23.147.18

Content breakdown by MIME type (Repeat View)



MIME Type	Requests
html	3
image	2
css	0
flash	0
font	0
js	0
other	0

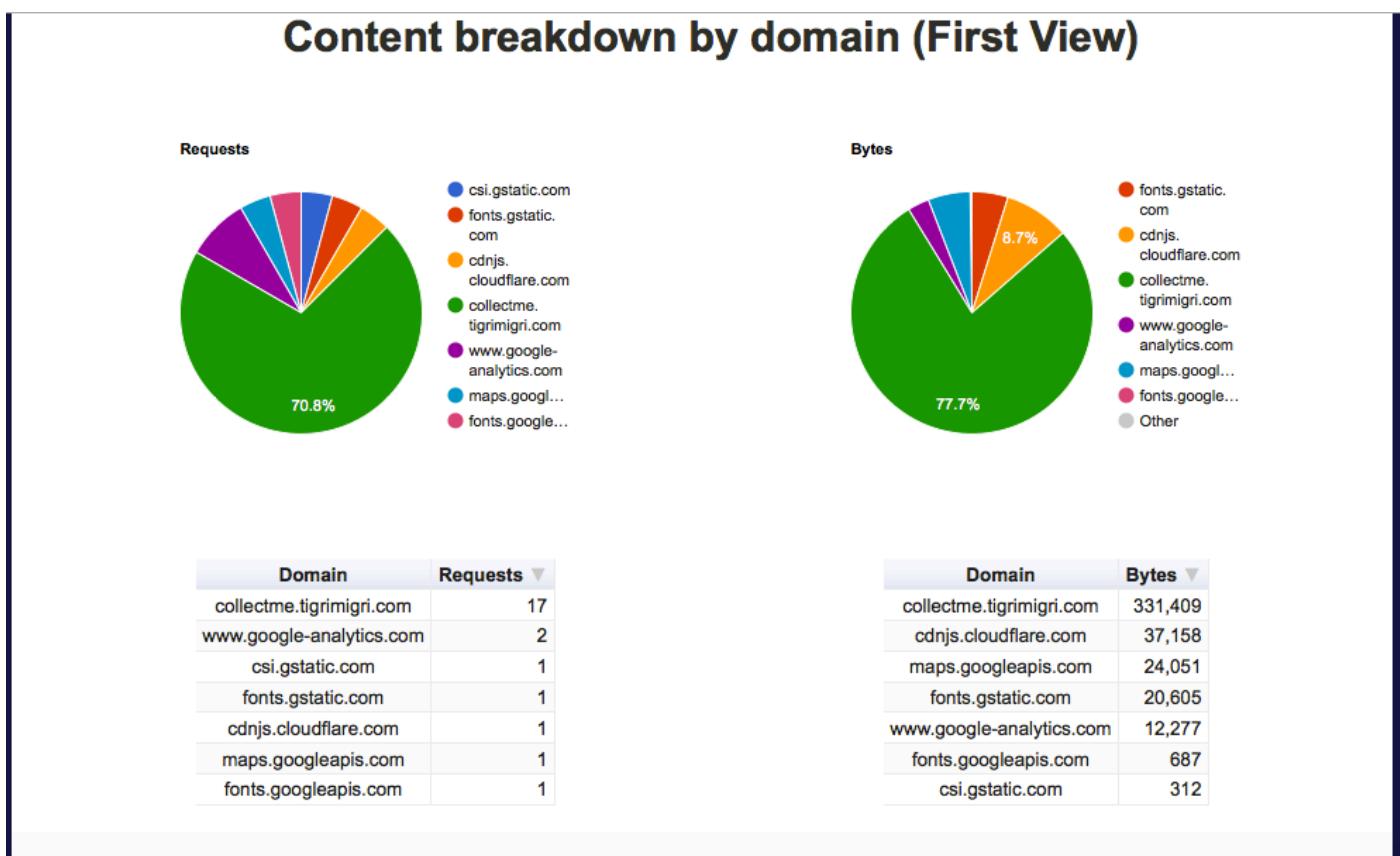
MIME Type	Bytes
html	4,659
image	696
css	0
flash	0
font	0
js	0
other	0

MAMP

Connection View (Repeat View)

Performance Review

Full Optimization Checklist						
	Keep-Alive	GZip	Compress Img	Progressive	Cache Static	CDN Detected
http://collectme.tigrimigri.com/index...	100%	100%	41%	0%	10%	22%
1: collectme.tigrimigri.com - index.html	✓	✓			✗	
2: collectme.tigrim... - angular-route.js	✓	✓			✗	✗
3: collectme.tigrim... bootstrap.min.css	✓	✓			✗	✗
4: cdnjs.cloudflare.com - angular.min.js	✓	✓			✓	✓
5: maps.googleapis.com - js		✓			✗	✓
6: collectme.tigrimigri.com - app.css	✓	✓			✗	✗
7: collectme.tigrim...modern-business.css	✓	✓			✗	✗
8: collectme.tigrim...ont-awesome.min.css	✓	✓			✗	✗
9: collectme.tigrimigri.com - jquery.js	✓	✓			✗	✗
10: collectme.tigrim... - bootstrap.min.js	✓	✓			✗	✗
11: fonts.googleapis.com - css					✗	✓
12: collectme.tigrimigri.com - alan.js	✓	✓			✗	✗
13: collectme.tigrimigri.com - status.gif	✓				✗	✗
14: csi.gstatic.com - csi						
15: www.google-anal....com - analytics.js	✓	✓			⚠	✓
16: collectme.tigrimigri.com - nav.html	✓	✓			✗	
17: collectme.tigrimigri.com - login.html	✓	✓			✗	
18: fonts.gstatic.c...ceF7z0kolaX1A.woff2					✓	
19: www.google-analytics.com - collect	✓					
20: collectme.tigr...i.com - iconFav.png	✓				✗	✗
21: collectme.tigr...lings-regular.woff2	✓				✗	✗
22: collectme.tigri....com - heading.json	✓				✗	
23: collectme.tigrimigri.com - family.jpg	✓		⚠	✗	✗	
24: collectme.tigrimigri.com - popup.png	✓				✗	
25: collectme.tigr...i.com - favicon.ico	✓	✓			✗	
	Keep-Alive	GZip	Compress Img	Progressive	Cache Static	CDN Detected



Hosting details:

Tigrimigri.com

Another hosting check:

Collectme.tigrimigri.com Is Hosted By [OVH Hosting](#)

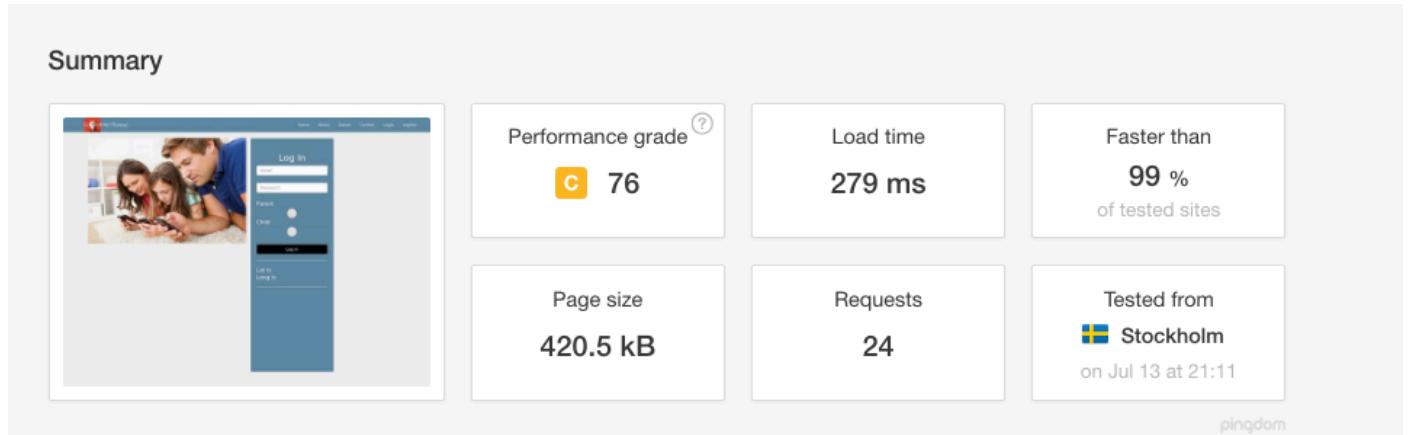


W3C® Markup Validation Service
Check the markup (HTML, XHTML, ...) of Web documents

1. **Error** Attribute `ng-app` not allowed on element `html` at this point.
From line 1, column 16; to line 2, column 34
`<TYPE html><html lang="en" ng-app="alanApp"><head>`
Attributes for element `html`:
[Global attributes](#)
2. **Error** Attribute `ng-include` not allowed on element `div` at this point.
From line 39, column 5; to line 39, column 42
`<div ng-include="'partials/nav.html'"></div>`
Attributes for element `div`:
[Global attributes](#)
3. **Error** Attribute `ng-view` not allowed on element `div` at this point.
From line 45, column 1; to line 45, column 17
`<div ng-view=""><img alt="S`
Attributes for element `div`:
[Global attributes](#)
4. **Error** An `img` element must have an `alt` attribute, except under certain conditions. For details, consult [guidance on providing text alternatives for images](#).
From line 45, column 18; to line 45, column 49
`-view="" ></div>`

Ping test

<https://tools.pingdom.com/#!/cjgxJW/http://collectme.tigrimigri.com/index.html#/login>

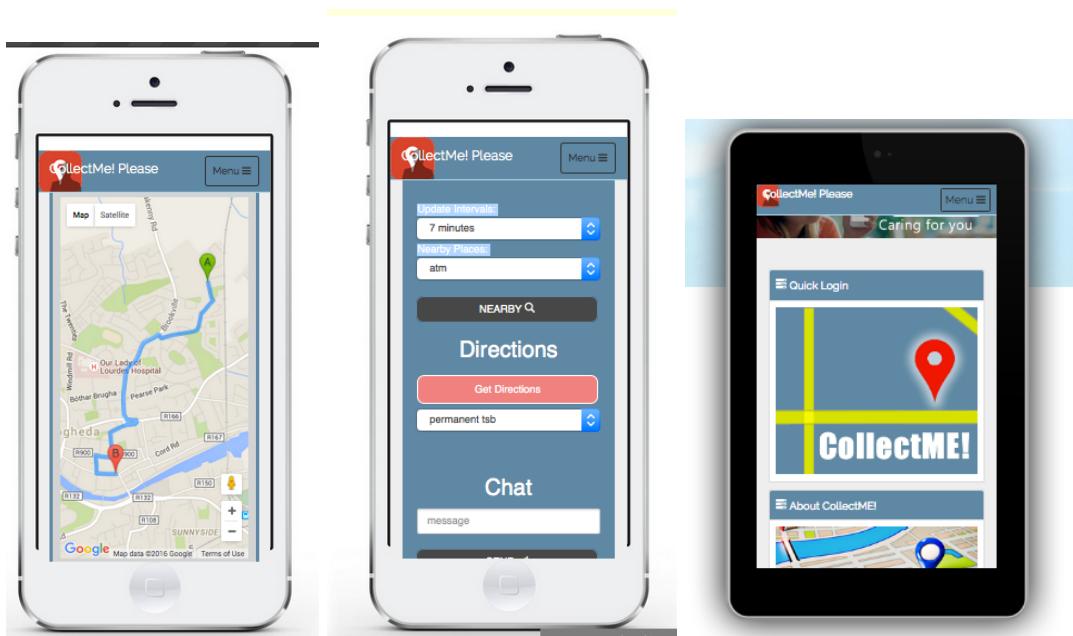


Ipad testing

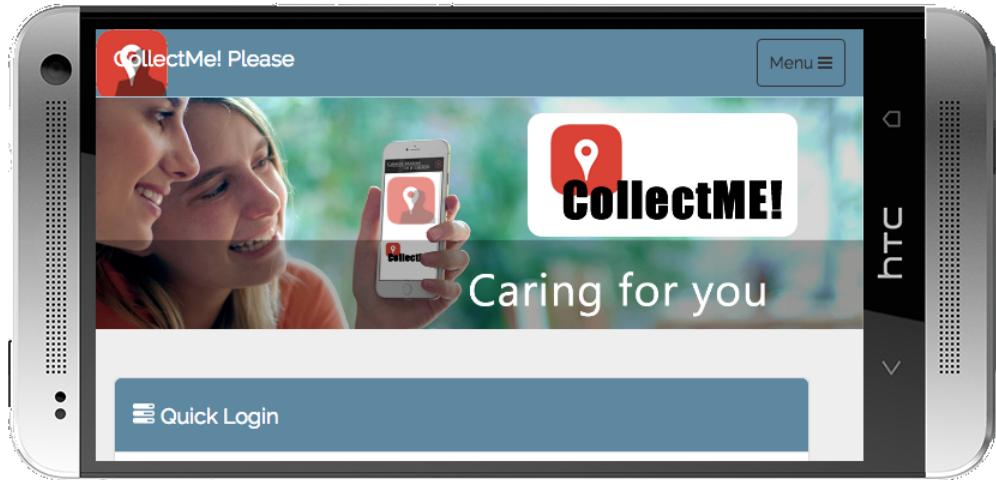
<http://www.responsimulator.com/?url=http%3A%2F%2Fcollectme.tigrimigri.com%2Findex.html%23%2Flogin#320>

Nexus 7

http://mobiletest.me/google_nexus_7_emulator/?u=http://collectme.tigrimigri.com/index.html#/



Requirements Specification



FILE	SIZE	0.0s	0.1s	0.2s	0.3s	0.4s	0.5s	
index.html collectme.tigrimigri.com/	1.3 kB	<div style="width: 10%;">10%</div>	<div style="width: 10%;">10%</div>					▼
angular.min.js cdnjs.cloudflare.com/ajax/libs/angula...	36.3 kB		<div style="width: 10%;">10%</div>					▼
js?key=AlzaSyAM2PLJqr7BNT-J0YgEEvuc5... maps.googleapis.com/maps/api/	23.5 kB		<div style="width: 10%;">10%</div>					▼
angular-route.js collectme.tigrimigri.com/js/	8.0 kB		<div style="width: 10%;">10%</div>					▼
{ } bootstrap.min.css collectme.tigrimigri.com/css/	20.7 kB	<div style="width: 10%;">10%</div>	<div style="width: 10%;">10%</div>	<div style="width: 10%;">10%</div>				▼
{ } app.css collectme.tigrimigri.com/css/	1.5 kB	<div style="width: 10%;">10%</div>	<div style="width: 10%;">10%</div>					▼
{ } modern-business.css collectme.tigrimigri.com/css/	1.0 kB	<div style="width: 10%;">10%</div>	<div style="width: 10%;">10%</div>					▼
{ } font-awesome.min.css collectme.tigrimigri.com/font-awesome...	5.2 kB	<div style="width: 10%;">10%</div>	<div style="width: 10%;">10%</div>					▼
{ } css?family=Raleway fonts.googleapis.com/	534 B	<div style="width: 10%;">10%</div>	<div style="width: 10%;">10%</div>					▼
status.gif collectme.tigrimigri.com/images/	3.1 kB		<div style="width: 10%;">10%</div>					▼
jquery.js collectme.tigrimigri.com/js/	32.8 kB		<div style="width: 10%;">10%</div>	<div style="width: 10%;">10%</div>	<div style="width: 10%;">10%</div>			▼
bootstrap.min.js collectme.tigrimigri.com/js/	9.7 kB	<div style="width: 10%;">10%</div>	<div style="width: 10%;">10%</div>	<div style="width: 10%;">10%</div>				▼
alan.js collectme.tigrimigri.com/js/	7.1 kB		<div style="width: 10%;">10%</div>					▼
csi?v=2&s=mapsapi3&v3v=25.8&action=ap... csi.gstatic.com/	0 B		<div style="width: 10%;">10%</div>	<div style="width: 10%;">10%</div>	<div style="width: 10%;">10%</div>			▼
lczWvq5y_Cwww_rBjOtT0w.woff fonts.gstatic.com/s/raleway/v11/	24.5 kB		<div style="width: 10%;">10%</div>		<div style="width: 10%;">10%</div>			▼

Requirements Specifications

Purpose

The purpose of this document is to set out the requirements for the development of collectMe app. This app will allow parents to keep track of their young teens when they go out with their friends to the cinemas, shopping centres or when out at theme parks and crowded public events.

The intended customers are parents of young teens aged between 12 – 15. The reason I targeted this age bracket is not to be an invasion of privacy for teenagers as a whole.

Project Scope

The scope of the project is to develop a workable app that can be used for the target audience. The project is laid down over the summer semester in NCI. Each week the project manager will deliver the required documentation to allow this app to grow and develop.

The app will have analysis, design, implementation, testing and evaluation phases. The system life cycles will work in iterations and each integration will lend and add to the last, each making the app better for the consumer.

.....

User Requirements Definition

User Story:

User: Mary wants to check where her daughter is. It is the first time Mary has let her daughter go out to the cinema with her friends.

CollectMe: Will allow Mary to go to the collect me web site and login or register as a new user. Mary can register herself and her daughter using the same email and passwords but making sure she is registered as a parent.

User: Mary is not used to computers and would like an easy to use interface

CollectMe: Collect me app will have a simple user interface. We will design with this in mind. A simple login that brings the user directly to the map page and this will show her daughters position at regular intervals.

User: Mary concern is that her daughter will login.

ColletMe: The app will allow Mary to send her daughter an email to reminder to login, or Mary can activate the application before her daughter leaves the house.

User: Mary would like to contact here daughter.

CollectMe: Collect me app has the ability to send emails or have a real-time chat.

User: Mary would like to collect her daughter.

CollectMe: The app is designed that when the child wants to be collected they press the “collect me” button and this alerts the parent of the time and positions of the child. It also sends an alternative email to the other parent or any guardians.

User: Mary would like to have a security alert or test to confirm that the child was collected safely.

CollectMe: will email any nominated guardians that the child was collect safely.

Requirements Specification

No requirements Specification are needed by the user to use the app. Collect me app's functionality will look after all the requirements from login to logout of the user.

Collect me app will use PHP, AngularJS, HTML5, CSS3, MySql database to create a functional application that is useful to the consumer.

Functional requirements

Login: The user logs in and shows their location. It also shows selected family members. The system contacts a database and checks to see if the user exists. If so the system will display a geographical representation of their family members whereabouts. Parent and child can login separately

Database: the system will use a database to store names, address and locations of each user. Some users will be children and others will be parents.

Google map: The system will depend on Google map API to gather the users locations, the System can also turn coordinates into actual street address as this may help if road works has diverted the driver and they must ask for directions.

Ajax calls: The app will make several calls to the database to update locations. This will give the user a real-time experience.

Collect me: The system will have a collect me button so that the parent can locate the child. The system will send alerts in the form of banners on the page, an email and a possible text message.

Security: The system will contact the parent to check that the child has been collect, if not the application will contact any guardians named.

Use Case Diagram

Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.

The Use Case Diagram provides an overview of all functional requirements.

Requirement 1 <Collect me button>

Requirement 1: Collect me button is the main feature of this app. The users will rely on they feature to work correctly. The system will focus on this fuctionallity in particular.

Description & Priority

The system must be able to carry out this function and keep working afterwards. It is a very important part of the system. The collect me button can also act as a panic button if the user finds themselves in distress.

Use Case part i

Actors Parent, child

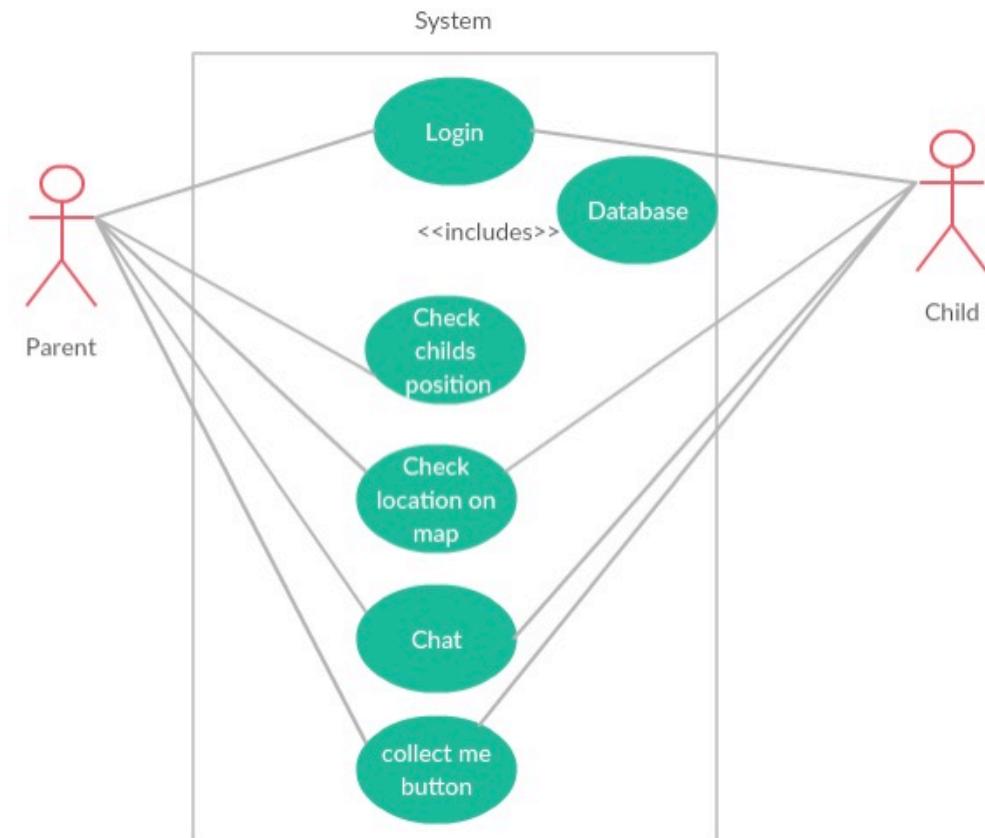
Scope

The scope of this use case is to ensure the 'collect me' functionality works.

Description

This use case describes how two users communicate together to send vital information. The collect me button is the main feature of the application. The Parent will be able to see the exact location of the child with the collect me. The child in turn can see their parent's movements towards them.

Use Case Diagram



Flow Description

Precondition

The system is working correctly and the users are tracking each others positions. They can chat or email each other. The system will be updating positions the whole time and displaying the information on Google maps with street address.

Activation

This use case starts when the Child clicks the "Collect me" button and the parent gets an alert, a text message; pop up on screen or an email.

Main flow

The user clicks the button "collect me" and the parent gets and alert. Both users can see each others position

1. The parent gets a alert sent by the system
2. The parent clicks to acknowledge the alert and moves to collect the child
3. The system sends an email to all parents to let all parties know that the child is due to be picked up. (this is in case of road accident or any other setback)
4. The child and parent meet, they tell the system that all is well,
5. Both users log out of the system.

Alternate flow

- 3: The child decides to stay a little longer with her friends
4. The system will still track the position of the child and send updates to the parent.
5. The parent can choose to keep looking at the child's position on the map or allow the application to alert them with an email or popup window
6. The child when ready will choose to press the collect me button.

Exceptional flow

E1 : Cancelled by the child

1. The system will alert the adult or parent that the child has cancelled the pickup.
2. The parent can make a decision to contact the child about new arrangements.
3. The system returns to normal tracking mode.

Termination

The users update the system and confirm that everyone has been collect and are safe.

The system will update any other guardians or parents that the child has been collected.

Post condition

The system goes into a wait state

Requirement 2 < Login and get coordinates >

Description & Priority

The login and gather user coordinates is another vital requirement for the system. The user logs in and the system together with google API captures the users exact whereabouts providing the user agrees to disclose their position. If they do not disclose their position the application is useless.

Use Case

Actors Parent, child

Scope

The scope of this use case is the login and collection of coordinates.

Description

This use case describes the how the user can login and allow their location to be collected and logged in a database.

Flow Description

Precondition

The system is running as a normal web page with a single page application that has information on the product.

Activation

This use case starts when a parent or child logs in and their coordinates are taken by the application

Main flow

1. The system identifies who the user is and displays their relevant information.
2. When the parent has logged in the system will display the whereabouts of their children. That is if the child has logged in also.
3. The database gathers the coordinates every 2 minutes and displays the information on Google map API
4. Both users can see each others position and this ensures the safety

Alternate flow

A1 : < Rejections of Coordinates >

1. The user rejects the systems request for their location.
2. The System reverts back to the single page application with a few static pages with information on the product
3. The user can contact the system administrator if they have any questions.

Exceptional flow

The system crashes or the users have no access to the internet

Termination

The user can choose to log out at anytime and the system stop all tracking.

Post condition

The system goes into a wait state

Non-Functional Requirements

Specifies any other particular non-functional attributes required by the system. Examples are provided below. **Remove the requirement headings that are not appropriate to your project.**

Performance/Response time requirement

The system will run a set time out request. This will run a MySQL query on the users data.

Security requirement

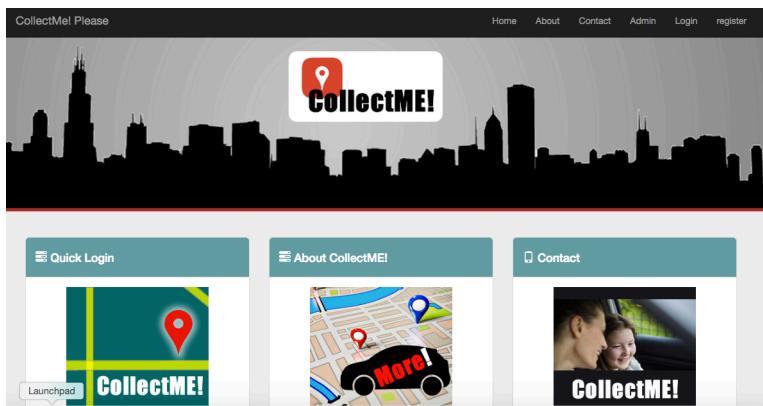
The system has security requirement built in. It is very important to protect the users who have registered with the application. It is also of utmost importance to keep the users location private.

Interface requirements

The most important part of the interface is that it must be mobile compatible. It is very important not to lose any functionality on the mobile slimmed down version. The main functionality must be visible and clear. Navigation should be easy to understand. Maps must display correctly especially in smaller size screens like phones and tablets.

The desktop version will also be user friendly and easy to use. With no distracting advertisements.

GUI (mock-up or prototype)



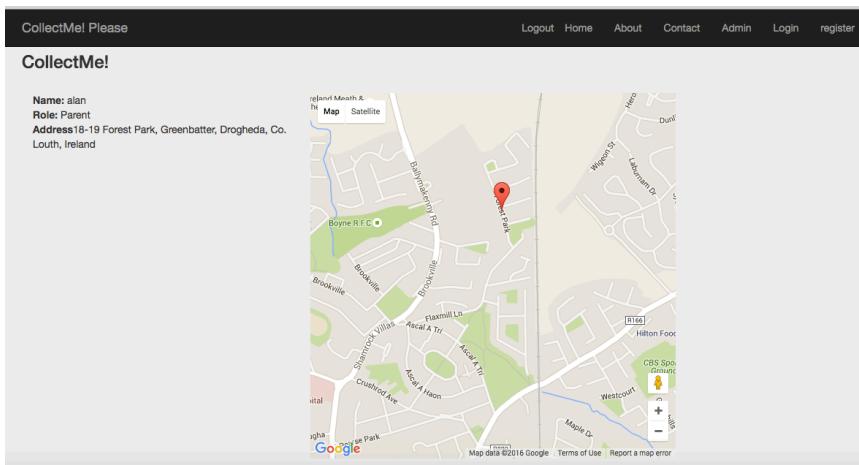
Static home page

The screenshot shows a registration form with a dark header bar containing the text "CollectMe! Please". Below the header is a large title "Register". Underneath the title is a message placeholder {{error_message}}. The form consists of two input fields: "Username" and "Password", followed by a "Register" button.

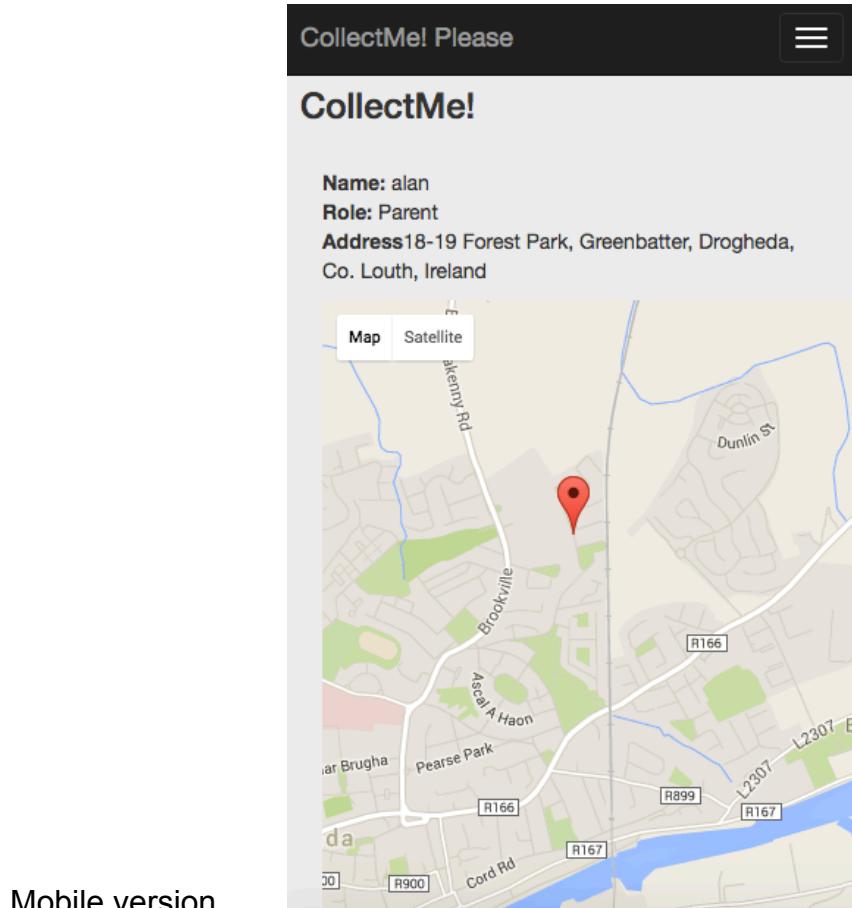
Static login page

The screenshot shows a contact form with a dark header bar containing the text "CollectMe! Please" and navigation links: Home, About, Contact, Admin, Login, and register. The main content area has a title "Contact CollectMe" and a breadcrumb trail "Home / Contact". It features a "Contact by email" section with fields for Name, Email, Subject, and Message, and a "Send Message" button. To the right is a "Contact Details" section with an email icon, a note about office hours (Monday - Friday, Anytime), and social media links for Facebook, LinkedIn, Twitter, and Google+.

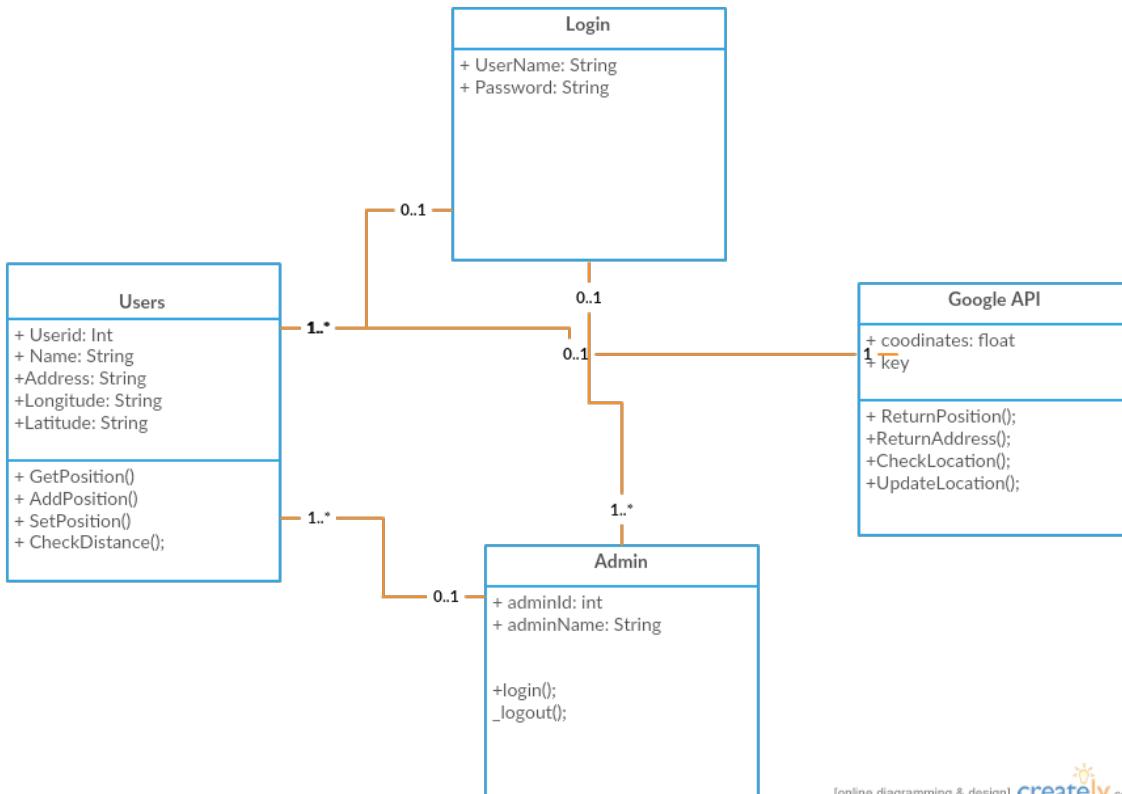
Static contact page



Main map page



System Architecture



[online diagramming & design]  [creately.com](#)

System Evolution

This section describes how the system could evolve over time.

Analysis and design documentation

Purpose of The Application Report Document

The Application Report document documents and tracks the necessary information required to effectively define architecture and system design in order to give the development team guidance on architecture of the system to be developed. The Application Report document is created during the Planning Phase of the project. Its intended audience is the project manager, project team, and development team. Some portions of this document such as the user interface (UI) may on occasion be shared with the client/user, and other stakeholder whose input/approval into the UI is needed.

General Overview and Design Guidelines/Approach

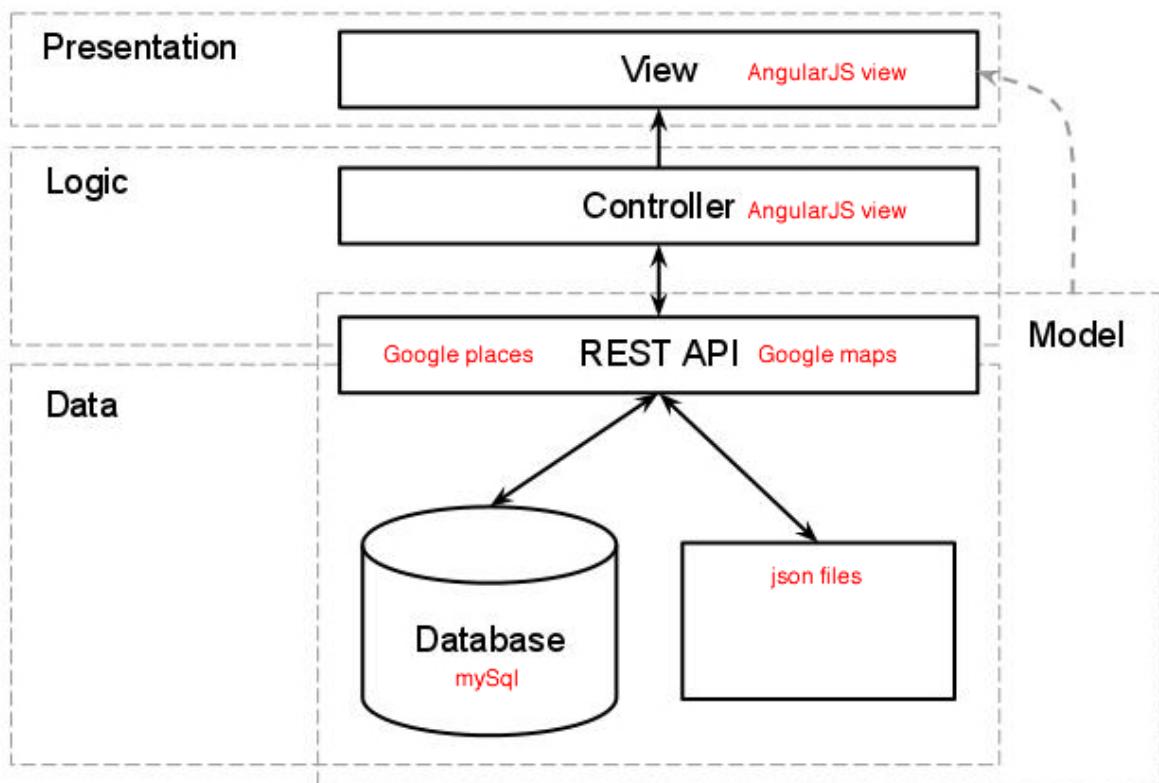
- Collect me applications will collect information provided by Google maps and Google places.
- The information will then be added to our database and display through an MVC angular application.
- The system relies on a strong Internet connection.
- The system will use MySQL as preferred database
- The application will work on all modern browsers and will be responsive.
- The Collect me application will use JavaScript set time out functions to update itself through specified time internals set by the user.
- The backend of this project will use php. This way it can easily connect to any MySQL database once the correct connection string is set.
-

Assumptions / Constraints / Standards

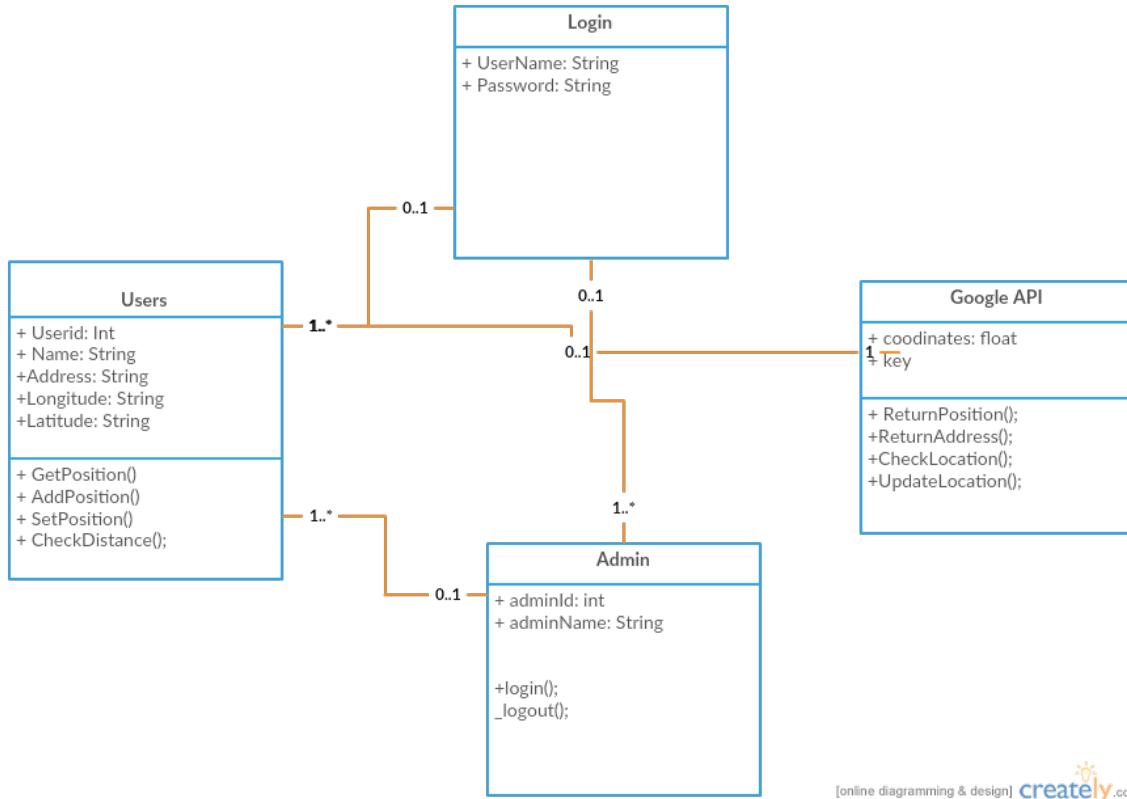
- A good internet connections is required to work this application
- The users of this application will be adults and children
- The child's functionality will be restricted
- The adult's functionality will be more advanced. They will have options like Google places and a flight path display to show the child's previous three locations

- The app will have an easy to use GUI and interface.

This section outlines the system and hardware architecture design of the system that is being built.

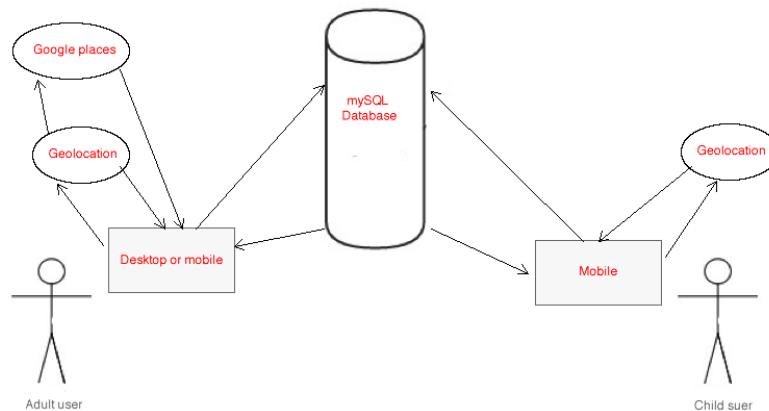


Logical View



[online diagramming & design]  [creately.com](#)

Software Architecture



Use-Cases – part II



Database Design

- Database is called “collectme”
- Customers table
- Positions table
- Messages table
- MySQL database
- Using phpMyAdmin as software

The screenshot shows the phpMyAdmin interface for the "collectme" database on "localhost".

Left Sidebar: Shows the database tree structure with databases: "alan_test", "collectme" (selected), and "information_schema". Under "collectme", there are tables: "customers", "message", and "position".

Top Bar: Includes tabs for Structure, SQL, Search, Query, Export, Import, Operations, and More.

Table View: A grid showing the structure of the three tables:

	Table	Action	Rows	Type
<input type="checkbox"/>	customers	Browse Structure Search Insert Empty Drop	16	MyISAM
<input type="checkbox"/>	message	Browse Structure Search Insert Empty Drop	144	InnoDB
<input type="checkbox"/>	position	Browse Structure Search Insert Empty Drop	3,230	InnoDB
3 tables		Sum	3,390	InnoDB

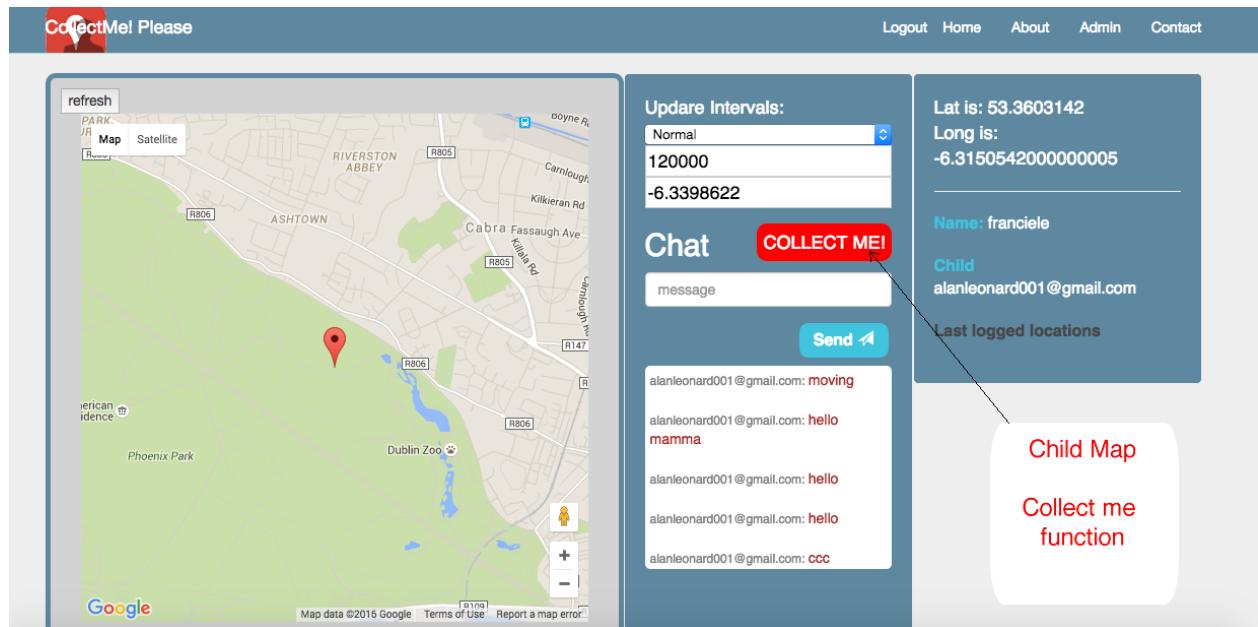
Bottom Navigation: Includes Print view and Data Dictionary links.

Data Conversions

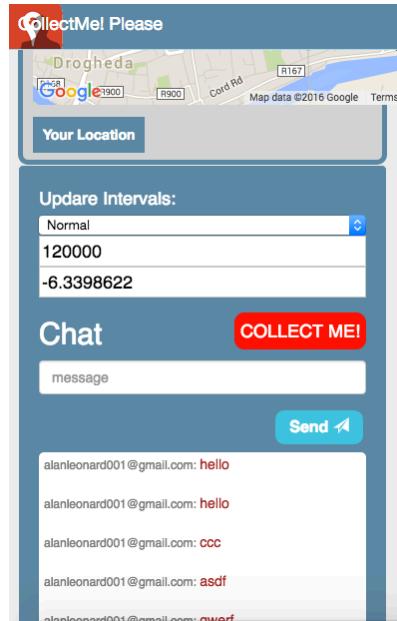
- N/A for this project

Application Program Interfaces

Child desktop



Child Mobile view



User Interface Design

The diagram illustrates the user interface design for the CollectMe! Please application, divided into several sections:

- Header and logo:** Located at the top left, featuring the application's name "CollectMe! Please".
- Main map:** A large map view showing a residential area with streets like Ballymakenny Rd, Wogans Build Centre, Forest Park, Forest Grange, and Ballypark. A blue line connects the user's current position (indicated by a blue dot) to another location.
- Options:** A menu bar at the top right with links to 'Logout', 'Home', 'About', 'Admin', and 'Contact'.
- Current pos:** Displays the user's current coordinates: Lat is: 53.728379 and Long is: -6.3397931.
- Last positions:** A list of recent locations visited by the user, including fran@mail.com: sdf, sdf, ddd, Please Collect me, and can you work.
- Last logged locations:** A list of the last locations logged, including fran@mail.com, Address 18-19 Current location Forest Park, Greenbatter, Drogheda, Co. Louth, Ireland, and Collect: 0.
- Chat:** A section for communication, containing a message input field, a 'Send' button, and a scrollable list of messages from 'fran@mail.com'.

Performance

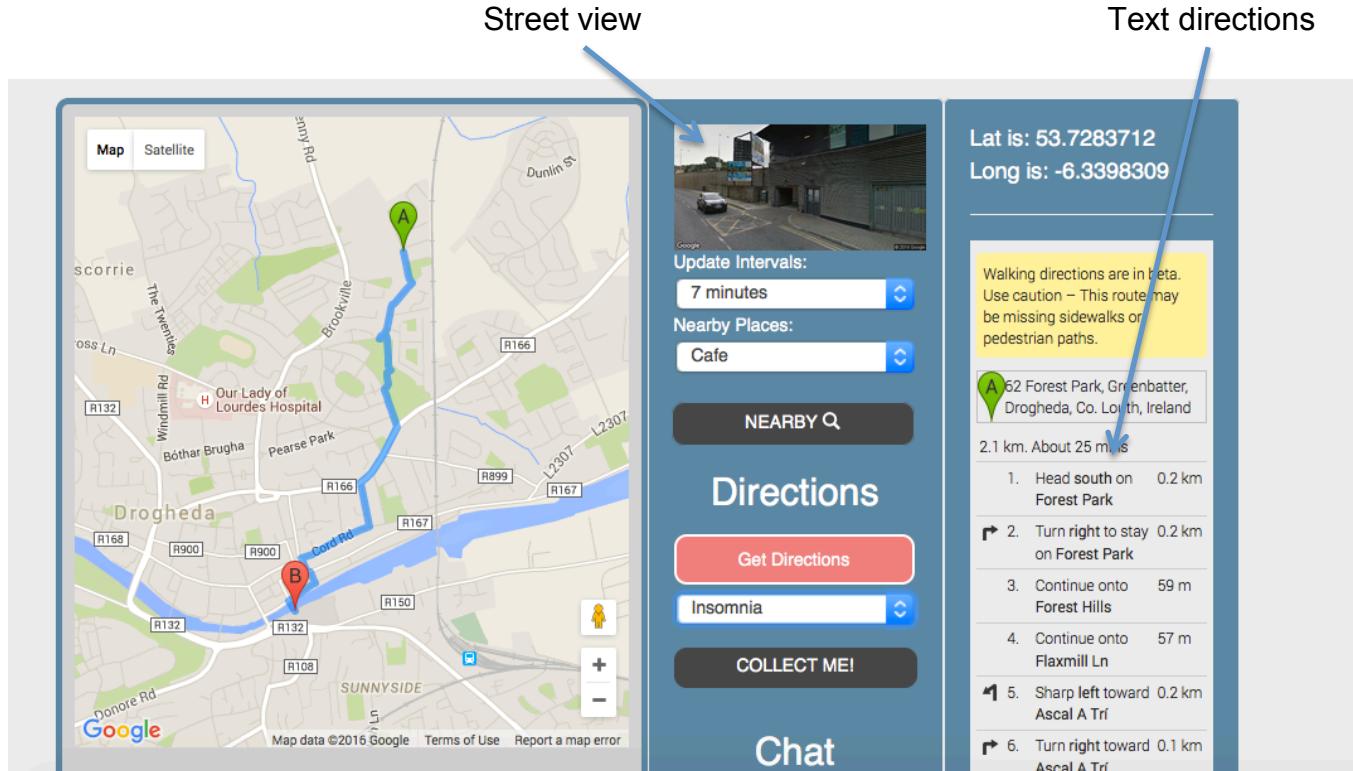
A strong Internet connection is needed. A set time out function will update the application. This can be set by the user or can run on the default time set.

Appendix B: Key Terms

The following table provides definitions for terms relevant to this document.

Term	Definition
Google Places	API provided by Google to gather information on current location
Geo location	Gather current location of user
AngularJS	JavaScript to provide an MVC and single page application

Extra features



References

"Introduction To Angular.Js In 50 Examples (Part 1)". *YouTube*. N.p., 2016. Web. 18 June 2016.

<https://www.youtube.com/watch?v=TRrL5j3Mlvo>

"Angularjs Tutorial". *W3schools.com*. N.p., 2016. Web. 30 June 2016.

<http://www.w3schools.com/angular/>

"PHP 5 Mysqli Functions". *W3schools.com*. N.p., 2016. Web. 1 July 2016.

http://www.w3schools.com/php/php_ref mysqli.asp

"Easy Form Validation In Angularjs With Ngmessages". *SitePoint*. N.p., 2015. Web. 2 July 2016.

<https://www.sitepoint.com/easy-form-validation-angularjs-ngmessages/>

"Maps API: Application Development & Analytics". *Google.com*. N.p., 2016. Web. 3 July 2016.

<https://developers.google.com/maps/documentation/directions/>

"Angularjs Tutorial Series: Part 5 – Creating A Dropdown Control For Your Site | Codementor". *Codementor.io*. N.p., 2016. Web. 4 July 2016.

<https://www.codementor.io/angularjs/tutorial/create-dropdown-control>

"Getting Started". *Google Developers*. N.p., 2016. Web. 4 July 2016.

<https://developers.google.com/>

"Google API Tutorial". *W3schools.com*. N.p., 2016. Web. 7 July 2016.

<http://www.w3schools.com/googleapi/>

"Google Places API Google Developers". *Google Developers*. N.p., 2016. Web. 8 July 2016.

<https://developers.google.com/places/>

"Google Maps API: Directions Service". *YouTube*. N.p., 2016. Web. 8 July 2016.

<https://www.youtube.com/watch?v=W0juXNFLd6w>

"Displaying Text Directions With Setpanel() - Google Developers. N.p., 2016. Web. 17 July 2016.

<https://developers.google.com/maps/documentation/javascript/examples/directions-panel>

"Google Static Maps API | Google Developers". Google Developers. N.p., 2016. Web. 24 July 2016.

<https://developers.google.com/maps/documentation/static-maps/>