# Final Year Project Work Log

November 13th:

* Created github repo
* Updated README file
* Uploaded Research documentation
* Uploaded UCLA – Use cases and Logical architecture diagrams

November 20th:

* Updated research documentation
* Created a hello world angular application

November 22nd:

* Presented idea to BOA

November 28th:

* Created 3 new components:
  + Navbar: this component will be on the top of every page the user sees with links to home, and a budget planner. The option to sign in will also eventually be on this component
  + Budget-planner: This will be the parent component for the form to create a budget plan
  + Data-visualisations: this will be the component that holds the graphs and displays data about average income and expenditure in Ireland.
* Added initial navigation using routerLink
* Installed angular material
* Links used today:
  + <https://angular.io/tutorial/first-app/first-app-lesson-02>
  + <https://www.reddit.com/r/Angular2/comments/11swgc3/help_angular_material_not_displaying_correctly_ng/>
  + <https://material.angular.io/components/toolbar/examples>
  + <https://medium.com/@sahilangrio/building-a-common-navbar-component-in-angular-a-step-by-step-guide-96833dbb2bcb>
* TODO:
  + Call the gov.ie API to make sure it works
    - <https://www.youtube.com/watch?v=DkB-Cm7AB3g>
  + Style the navbar
  + Start designing the budget-planner

November 29th:

* Blockers:
  + Angular does not allow script tags in the html pages. It strips the tag itself an renders it useless
    - I could create a directive?
    - I tried to follow https://medium.com/ngconf/dynamically-execute-script-in-an-angular-component-e4dbcf9a34
    - Trying to input:

@code

```

<script src="https://cdn.jsdelivr.net/gh/CSOIreland/PxWidget@2.6.1/js/isogram.min.js"></script>

<script>pxWidget.queue('chart', 'pxwidget8739094941', {"autoupdate":true,"matrix":null,"type":"line","showPrecentage":true,"copyright":true,"link":"https://data.cso.ie/table/EHQ03","sort":false,"metadata":{"xAxis":{"TLIST(Q1)":["20233","20232","20231","20224","20223","20222","20221","20214","20213","20212","20211","20204","20203","20202","20201","20194","20193","20192","20191","20184","20183","20182","20181","20174","20173","20172","20171","20164","20163","20162","20161","20154","20153","20152","20151","20144","20143","20142","20141","20134","20133","20132","20131","20124","20123","20122","20121","20114","20113","20112","20111","20104","20103","20102","20101","20094","20093","20092","20091","20084","20083","20082","20081"],"role":"time"},"fluidTime":[0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,60,61,62],"api":{"query":{"url":"https://ws.cso.ie/public/api.jsonrpc","data":{"jsonrpc":"2.0","method":"PxStat.Data.Cube\_API.ReadMetadata","params":{"matrix":"EHQ03","language":"en","format":{"type":"JSON-stat","version":"2.0"}},"version":"2.0"}},"response":{}}},"data":{"labels":[],"datasets":[{"label":"avg weekly earning","pointRadius":2,"pointHoverRadius":4,"maxBarThickness":90,"api":{"query":{"url":"https://ws.cso.ie/public/api.jsonrpc","data":{"jsonrpc":"2.0","method":"PxStat.Data.Cube\_API.ReadDataset","params":{"class":"query","id":["STATISTIC","C02665V03225","C02397V02888"],"dimension":{"STATISTIC":{"category":{"index":["EHQ03C02"]}},"C02665V03225":{"category":{"index":["-"]}},"C02397V02888":{"category":{"index":["-"]}}},"extension":{"language":{"code":"en"},"format":{"type":"JSON-stat","version":"2.0"},"matrix":"EHQ03","m2m":false},"version":"2.0"}}},"response":{}},"data":[],"unit":[],"decimal":[],"fluidTime":[],"fill":false}],"null":".."},"options":{"responsive":true,"maintainAspectRatio":false,"title":{"display":true,"text":["Average Earnings, Hours Worked, Employment and Labour Costs"]},"tooltips":{"mode":"index","callbacks":{}},"hover":{"mode":"nearest","intersect":true},"scales":{"xAxes":[{"ticks":{"beginAtZero":false,"maxTicksLimit":null,"reverse":true},"gridLines":{"display":false},"scaleLabel":{"display":true,"labelString":"Quarter"},"stacked":false}],"yAxes":[{"display":true,"position":"left","id":"y-axis-1","ticks":{"beginAtZero":false},"callback":null,"scaleLabel":{"display":false,"labelString":null},"stacked":false}]},"plugins":{"stacked100":{"enable":false,"replaceTooltipLabel":false},"colorschemes":{"scheme":["#405381","#5BC1A5","#FCBE72","#0091AB","#90989F","#3D5999","#C6CC5C","#579599","#FBAA34","#007780","#B9BEC3","#00AF86","#6BC2C2","#00758C","#FCB053","#6A7794","#A3CCC1","#F68B58","#36A0B3"]}},"legend":{"display":true,"position":"bottom"},"elements":{"line":{"tension":0.4}},"updated":"2023-11-28T11:00:00Z","layout":{"padding":{"left":22,"right":22,"top":0,"bottom":22}}},"datasetLabels":["avg weekly earning"],"plugins":[{}]});</script>

<div id="pxwidget8739094941" class="pxwidget"></div>

```

* + - From <https://data.cso.ie/table/EHQ04>

January 16

* Added all the form fields for the application
* Added styling for the pages

January 18

* Added `chart.js`
* Currently when a user fills out the whole form and presses submit, a doughnut chart appears at the end of the page and displays all the income and expenses

January 21:

* Styled rows
* Added a dropdown menu for whether the value is [weekly, monthly, yearly]
* Removed income from the doughnut graph
* Calculate total expenditure
* Calculate total income
* Calculate total surplus
* Chart is showing the wrong colours in the legend

January 28th:

* Attempted to add the script tags in from CSO, followed:
  + <https://blog.davidjs.com/2018/09/insert-script-tags-in-angular-components/>
  + <https://www.npmjs.com/package/ngx-script-loader>
  + <https://stackoverflow.com/questions/34489916/how-to-load-external-scripts-dynamically-in-angular>

Current Blockers:

* The doughnut chart displays the wrong colours in the legend
* Angular strips script tags so we need to find a way around it

Next Steps:

* Set up DB and write to it using backend

Stretch Goals:

* Users can name and input their own custom expenses