Contents

[**Initial Project planning:** 2](#_Toc26374695)

[**Meeting Log:** 3](#_Toc26374696)

[**Examples of group Communication:** 4](#_Toc26374697)

[**Measuring user contribution:** 5](#_Toc26374698)

[**Project plan – Gannt Chart:** 6](#_Toc26374699)

[**Task Delivery Form** 7](#_Toc26374700)

# **Initial Project planning:**

|  |  |
| --- | --- |
| User | Admin |
| Make claim | Add membership |
| Pay annual fee | View membership requests |
| Log in | Suspend members |
| Register (extension of log in) | View outstanding balances |
| View balance | Charge members |
| Pay outstanding balance (extension of view balance) | Claim eligibility assessment |
|  | Log in |

**User:**

* Make claim
  + User should be able to submit a claim after the first 6 months with a max of 2 per year
* Pay fees
  + A user should be able to pay the annual fee
* Log in
  + The user will have to be able to log into the system
* Register
  + If the user is a non-member, they should be able to register to become one
* View balance
  + The user should be able to view any outstanding balances
* Pay outstanding balances
  + If there is an outstanding balance, then the user should be able to pay

**Admin:**

* Add member
  + Once a membership request has been sent the admin should be able to upgrade them to a full membership
* View membership requests
  + The admin should be able to view a list of any membership requests
* Suspend members
  + The admin should be able to temporarily suspend a user’s membership if fees have not been paid
* View outstanding balances
  + The admin should be able to view a list of any outstanding balances
* Charge members
  + The admin should be able to charge the users based on an annual lumpsum and membership fee
* Claim eligibility assessment
  + The admin should be able to assess the eligibility of the members for the outstanding claim

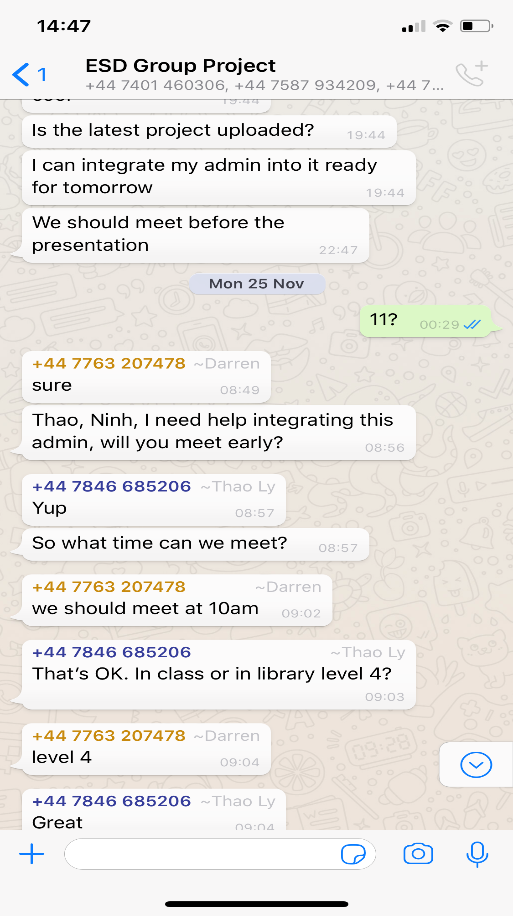
**Task identification:**

* Create a login page using jsp.
* Link login page to the login servlet to check if entered details match the users entered details.
* Create a class to handle login status with sessions.
* If login is verified then mark user as logged in using the session handler
* Identify tasks for the sprints and the who will be doing them.
* Make sessions lasting 20 minutes
* Create users table.
* Create payments table.
* Create members table.
* Create claims’ table.
* Create database access classes for users table.
* Create database access classes for payments table.
* Create database access classes for members table.
* Create database access classes for claims table.
* Create a user login page jsp file.
* Add registration to login page.
* Create a dashboard page for members user
* Create a view for admin to see daily activities.

# **Meeting Log:**

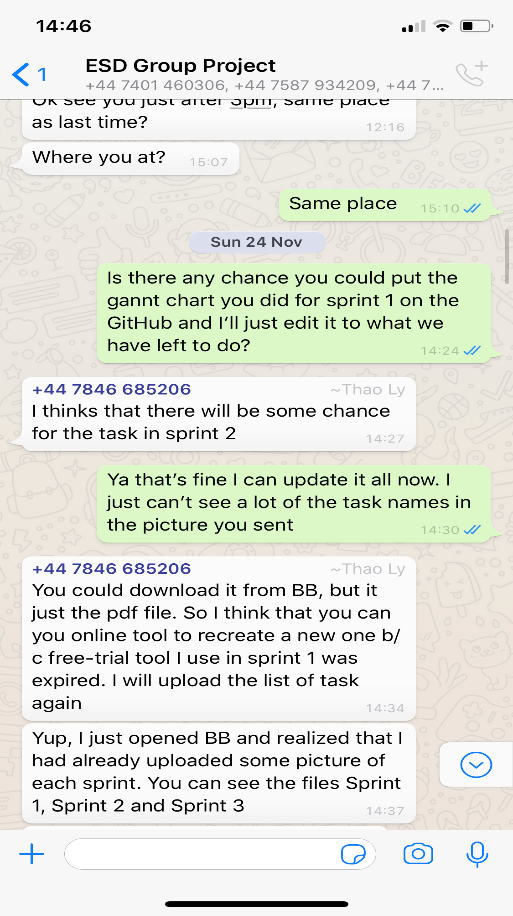
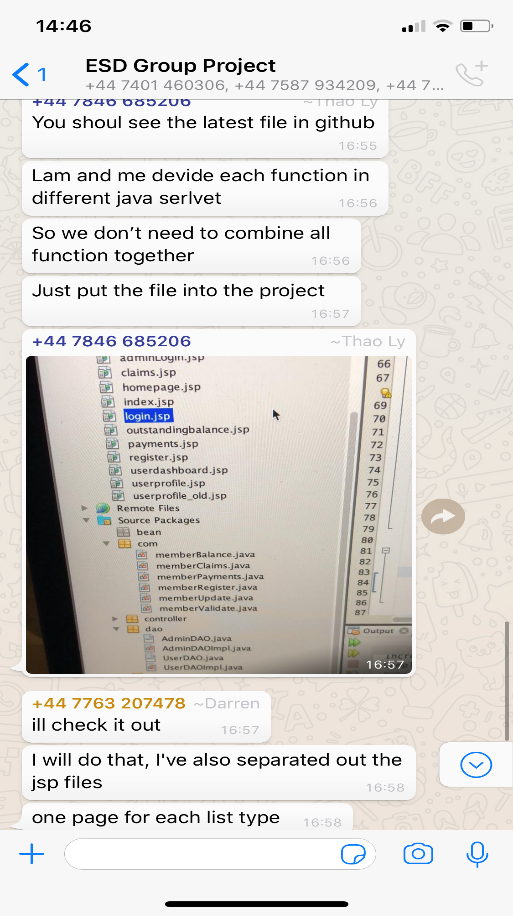
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Meeting 1** | **Meeting 2** | **Meeting 3** | **Meeting 4** | **Meeting 5** | **Meeting 6** | **Meeting 7** |
| **Date** | 7/11/19 | 11/11/19 | 14/11/19 | 22/11/19 | 25/11/19 | 28/11/19 | 04/12/19 |
| **Attendance** | Thao, Lam, Fintan, Darren | Thao, Lam, Fintan | Thao, Lam, Fintan, Darren, Nep | Thao, Lam, Fintan, Darren | Thao, Lam, Fintan, Darren | Thao, Lam, Fintan, Darren | Thao, Lam, Fintan, Darren |
| **Topic** | What has to be done | What each person has done | Allocating tasks for sprint 2 | What has been done and still to do | Go over everything and present sprint 2 | Allocating tasks for final sprint | Combine final product and test |
| **Duration** | 43 mins | 20 mins | 50 mins | 1 hour | 25 mins | 45 mins | 1 hour |

# **Examples of group Communication:**

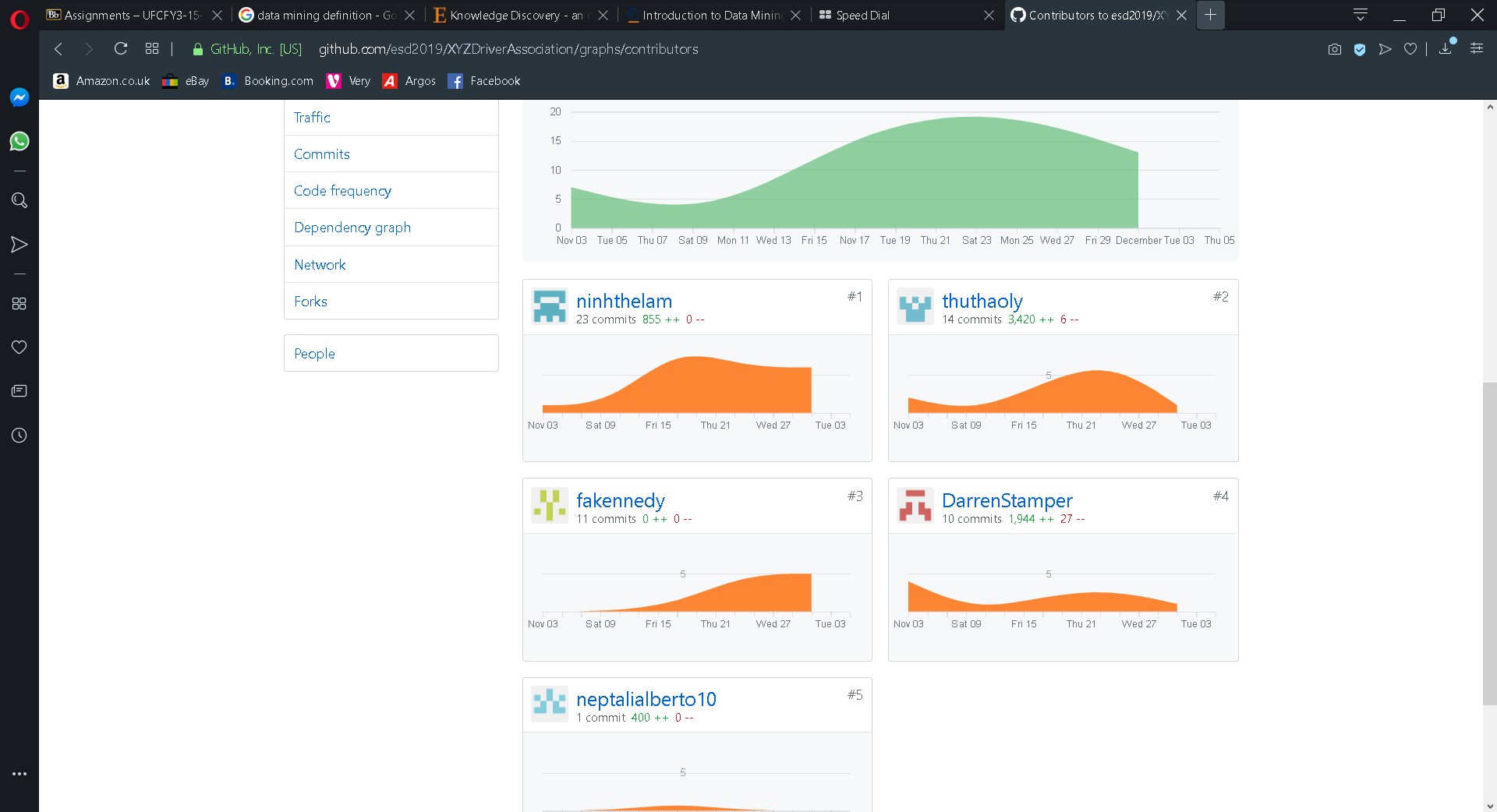
Below we have attached examples of the communication between all the group members via the Whatsapp group created. We predominately used this form of method to communicate any issues or tasks to be completed. We decided that this was the best way of communicating as each member already had this application saved onto their phones and used it regularly. The first image(right )is an example of the group setting up a meeting to discuss ideas and what has already been done. This message was based around the meeting on the 25/11/19 at 11am, discussing the structure of the presentation. We also integrated the admin into the whole system as we were having problems with it which was discussed in the sprint as well.

The second example shown (below on the right-hand side) is a member of the group asking for a piece of already completed work for editing. One member asked for the completed Gannt charts from the first hand in date so it can be updated. This was then placed onto the group wiki on blackboard.

The third example (below on the left-hand side) is a member of the group making everyone else aware that the latest updates of code have been placed onto github. They also gave advice on how they completed such code and what should be done. He also attached a picture to help explain his point.

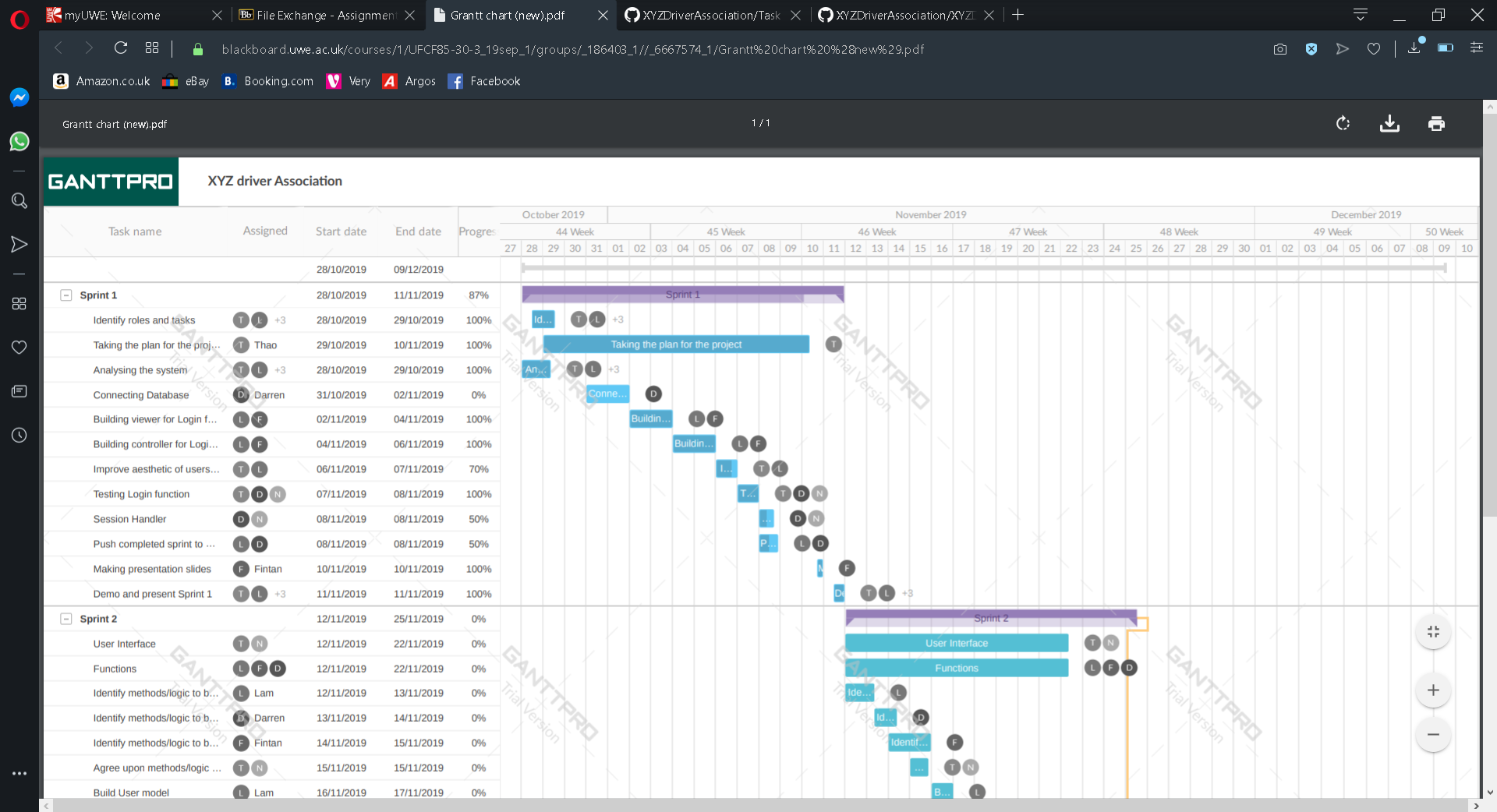
 

# **Measuring user contribution:**

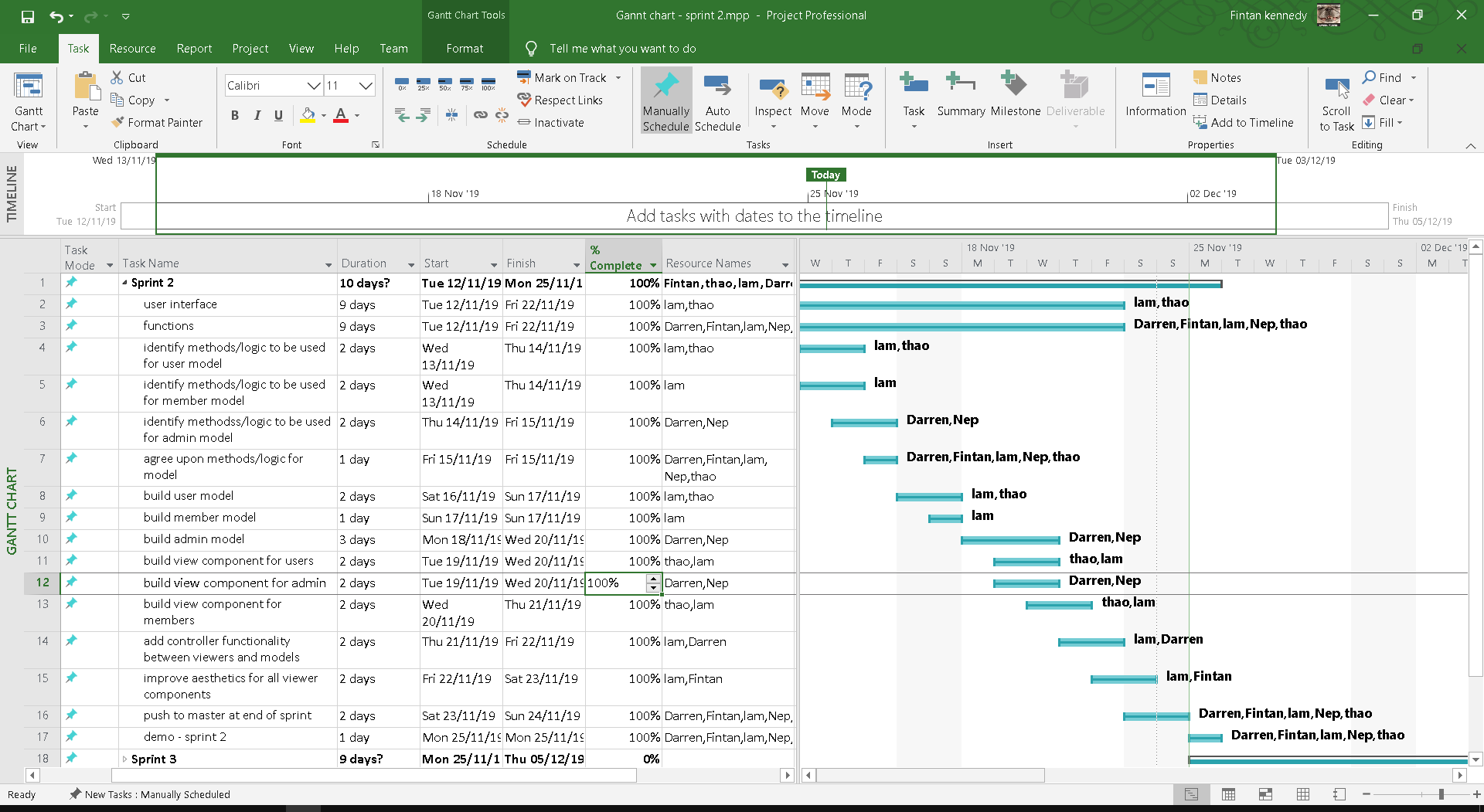


# **Project plan – Gannt Chart:**

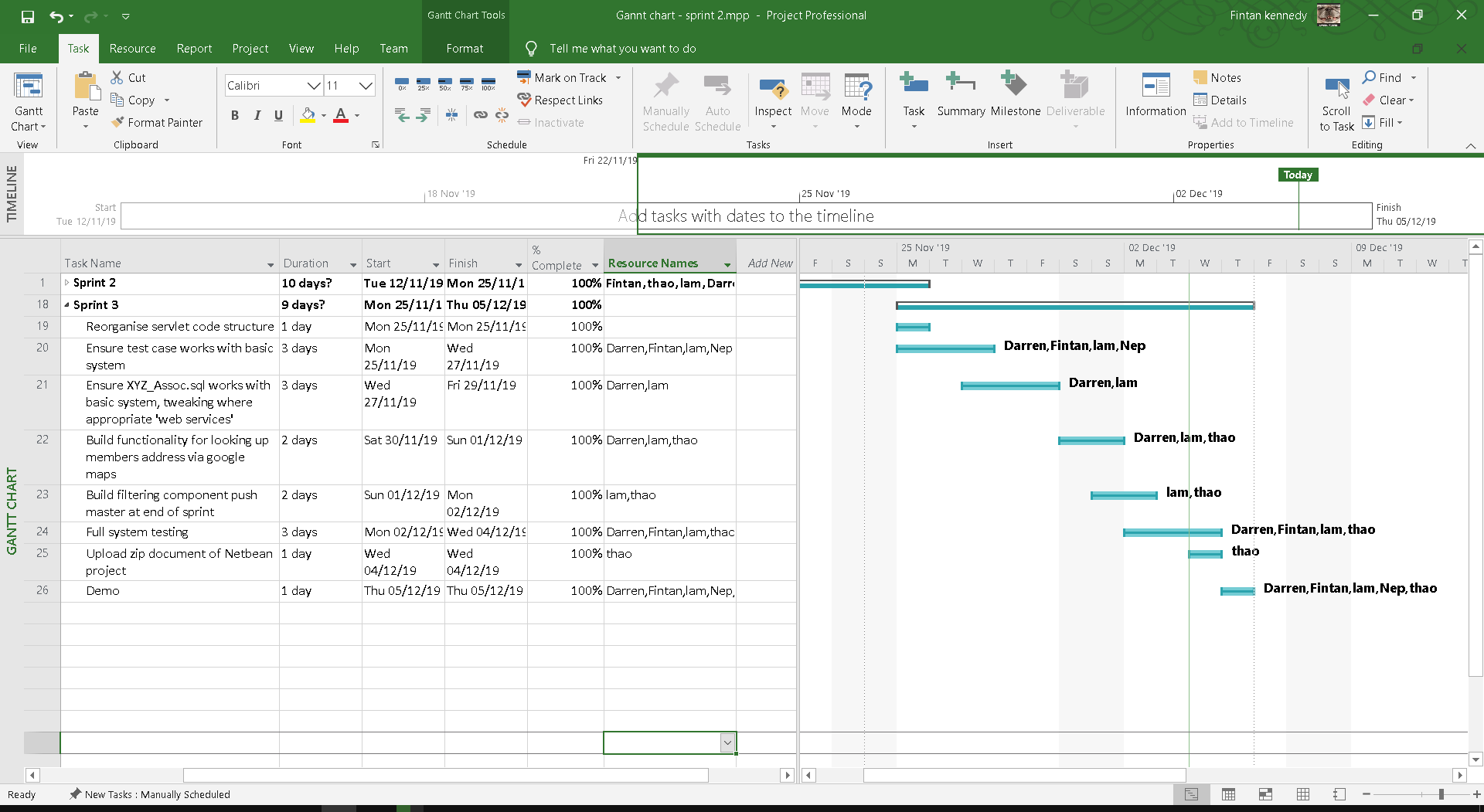
**Sprint 1:**

****

**Sprint 2:**



**Sprint 3:**



# **Task Delivery Form**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Tasks | | | | | | | | |
| Group Members | T1 | T2 | T3 | T4 | T5 | T6 | T7 | T8 | Signature | |
| Fintan Kennedy |  | ✓ |  |  |  | ✓ |  | ✓ |  | |
| Neptali Ortega-Gutierrez |  |  |  |  |  |  |  | ✓ |  | |
| Darren Stamper | ✓ | ✓ | ✓ |  |  | ✓ |  | ✓ |  | |
| Lam Ninh | ✓ | ✓ |  |  | ✓ | ✓ |  | ✓ |  | |
| Thao Ly |  | ✓ |  |  | ✓ | ✓ | ✓ | ✓ |  | |
| Status of Completion (%) | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |

**List of Task:**

1. Reorganise servlet code structure
2. Ensure test cases work
3. Ensure XYZ\_Assoc.sql works with basic system, tweaking where appropriate 'web services'
4. T4. Build functionality for looking up members address via google maps
5. Build filtering component push master at end of sprint
6. Full system testing
7. Upload zip document
8. Demonstrate