Progress Reports

# Progress Report for 19th March

## Week One

The project assigned requires us to develop a piece of software that serves as a dashboard capable of visualising data in a meaningful way and allowing the user to manipulate the given data. This progress report was written two days after our first meeting and its purpose is to provide an insight into how development of this project is coming along.

As we have only had one day since our first meeting we were still able to go over our project specification and identify who the stakeholders of the software would be, we also discussed some requirements that the software would have to fulfil as well as some resources that we would require in order to properly develop this software.

During our meeting we dedicated some time into opening the archive files given to us and conducting research to get an early idea of what type of data we would have to work with throughout this project.

## Decision Table

|  |  |  |
| --- | --- | --- |
| Description | Decision | Priority |
|  |  |  |
| Design Methodology | We have chosen to follow the agile development methodology.  It is preferable to each group member as we wish to complete tasks in bursts using an incremental and iterative method. | Neutral |
|  |  |  |
| Programming Language | Our choices were between Cocoa, C++, Java or Visual Basic.  Our final decision will be based on ease of use and its ability to support a GUI, for now it is undecided as it is not a high priority at this point in time. | Low |

## Issues

A large issue we had was not being able to have a group formed until the third week of semester, leaving us only four weeks to complete all our tasks until the first deliverable rather than 6. This means that we will need to organise frequent and extended group meetings in order to produce a quality report in time.

## Action Items

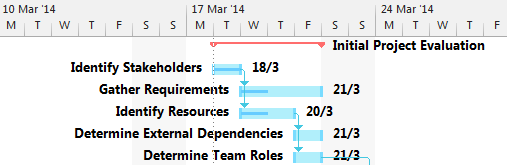
### Completed

No action items had been set the previous week

### Assigned

Action items that were assigned at the end of the meeting that we planned to have completed by the next week included a design plan, business case, an initial SRS, measuring effectiveness, milestones, risk analysis and a git repository created for the project.

## Current Status



The Gantt chart does not show us much regarding how much has been completed since we have only had one day to put work into the task.

# Progress for 26th March

## Week Two

This week we had all members present during the Tuesday tutorial so we were able to set a weekly date that we can all get together and simultaneously work on this project. The date chosen that suited us best was on Wednesdays at 5:30pm.

We were able to meet with the client and we gathered some requirements including security details, database updating, and user details. These are taken into more detail within the SRS document.

Peter has chosen to handle development on the front end of the software. Meaning he will handle the website and data visualisation while the others focus on the technical back end tasks.

Our team roles are as follows:

* James: Project Manager/Software Architect
* Kurt: Tool Specialist/Software Engineer
* Josh: Database Administrator
* Peter: UI Engineer/System Analyst
* Jamie: Software Engineer/Analyst

## Decision Table

|  |  |  |
| --- | --- | --- |
| Description | Decision | Priority |
|  |  |  |
| RDBMS\* | We have chosen to use MySQL as our RDBMS due to its capabilities of supporting concurrent updating across multiple clients since our software will need to support many users at once. | Neutral |
|  |  |  |
| Team Roles | We all discussed preferable parts of the project that we would like to be a part of and were able to determine team roles from this. From doing this we were able to assign tasks to people with the role that would best suit them. Those roles are shown in this report. | High |
|  |  |  |
| Programming Language | We revisited this decision as we have looked into developing the software to be based inside a browser. This gave us two options of languages being PHP, JavaScript and Ruby. JavaScript has a tool named D3js tool that can be used to show graphs and visualise data aesthetically. However most of us decided that Ruby would be the best language for us to develop in as it also has many tools available and it is a simple language for us to learn. | Neutral |

\*RDBMS: Relational Database Management System

## Issues

We had a small issue with communication where two members of our group had left early before our meeting with the client. However we were successfully able to complete our meeting with only three of us and gather more essential requirements.

## Action Items

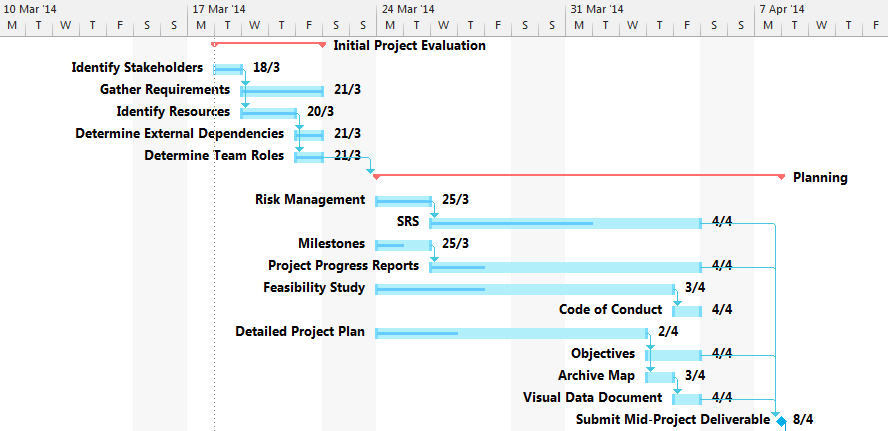
### Completed

Through further discussion regarding development of the project we were able to come up with a design plan. A risk management report was also completed during the week and the milestones were decided on through creation of the Gantt chart. A git repository was also created with each team member added as collaborators.

### Assigned

Jamie and Kurt were assigned the feasibility study, Peter and Kurt were assigned the SRS, James and Josh were assigned the detailed plan. Everybody was advised to look into the Ruby language in their own time to become more familiar with what we will be coding with.

## Current Status



Once we were given team roles we were than able to work on our given tasks. We are now currently working on the SRS, Progress reports (this document), feasibility study and a project plan. The milestones document still requires completion.

# Progress for 2nd April

## Week Three

This week we spent time looking through the Python SVN, bugs website and archive files for bug reports, test cases, execution traces/logs, emails, discussions and emails. We were only able to find the bug reports and emails for now. We are searching for this data so that when we get to our import data stage we would immediately know where the data would come from, thus saving time for ourselves in the future. Peter and Josh also conducted a meeting with our client and received more requirements for the SRS.

The Wednesday meeting consisted of checking up with each other’s progress and helping each other out so we could gain a mutual understanding of different parts of the project.

## Decision Table

|  |  |  |
| --- | --- | --- |
| Description | Decision | Priority |
|  |  |  |
| How to import data | We decided to use Ruby’s integrated unarchiver to access the data supplied to us in the Python archive files. Importing data is an integral part of our project which was why it was classed as a high priority task. | High |
|  |  |  |
|  |  |  |

## Issues

We had an issue with locating the data in SVN and the Python Archives. As we still had current tasks to complete before the deliverable we decided to put this issue on hold for a week while we focused on our report.

## Action Items

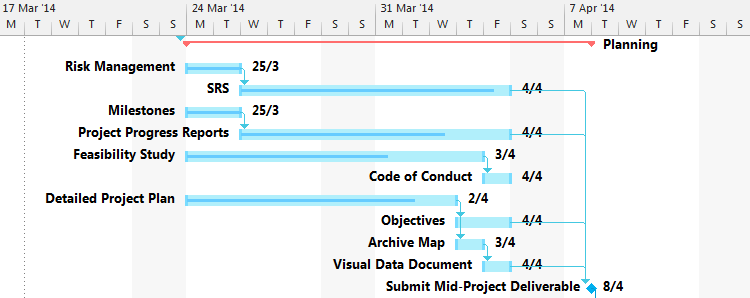
### Completed

Action items from last week required more than a week to complete, hence why no items have been completed. Members also studied Ruby during the week.

### Assigned

A code of conduct is to be completed by Jamie, Objectives were assigned to James, completion of an archive map was assigned to Josh, Peter and Josh were also assigned to have a visual data document completed.

## Current Status



We have one week until our deliverable is due and each team member is still working on the same task as the previous week. The milestones document has also been completed.

# Progress for 7th April

## Week Four

This is the last week before our deliverable is due, therefore all tasks must be completed before the 8th of April (tomorrow).

All sections of the mid-project deliverable have been complete and we just have to format the document and prepare it for submission.

## Decision Table

|  |  |  |
| --- | --- | --- |
| Description | Decision | Priority |
|  |  |  |
| Design Methodology | We originally decided on agile development however we have specifically chosen feature-driven development. We feel that this would suit our project as we have design our development phases into iterations. | Low |
|  |  |  |

## Issues

Because of the first couple of weeks being behind on meetings and progress, we decided that we would host a secondary meeting on the Friday to ensure that all deliverables would be completed.

## Action Items

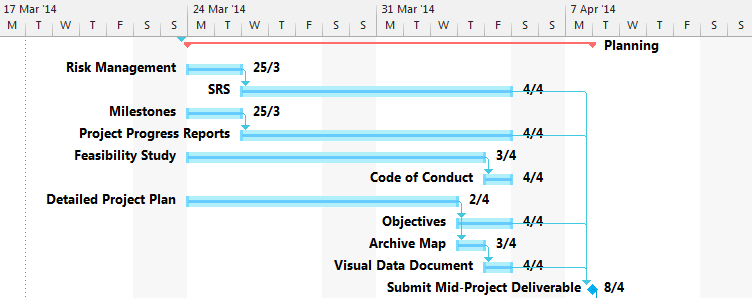
### Completed

All action items from last week have been completed in time for the deliverable.

### Assigned

Josh has been assigned to set up our MySQL database, James has been assigned to set up our code development environment, Jamie has been assigned to locate the required data from SVN and the Python archives while Peter has been assigned to set up tools required to host a website and begin development on that.

## Current Status



All tasks within planning phase of the project is complete and we are prepared to submit our report for the next milestone on the 8th of April. We are now ready to prepare for our execution stage and setting up the development environment.

# Progress for 16th April

## Week Five

After completing the mid deliverable and having a couple of days break, we decided to spend some time solely towards setting up our development environment. The reason for this was to ensure that each of us would have the same versions of programs and tools, thus preventing errors that may arise in the future due to incompatibility between software versions.

## Decision Table

|  |  |  |
| --- | --- | --- |
| Description | Decision | Priority |
|  |  |  |
| Database Editing Software | Sequel Pro 1.0.2  We decided this because it’s a quick and easy solution to setting up a local database and working with SQL files. | Low |
|  |  |  |

## Issues

One issue we had with setting up the development environment was that Peter and Josh both use Windows PC’s for development whereas James, Jamie and Kurt use Apple Macs. This meant that we had to separate different tasks amongst different members to avoid incompatibility. We also had to go through who would have what software for each different operating system.

## Action Items

### Completed

Josh has set up a MySQL database that we are able to use and we have not yet had the data imported from the SVN.

### Assigned

Jamie, Kurt and James should look into importing data this week and find methods we can use to implement that data into our program. Peter also needs to set up a basic website which will be used to host our product throughout this project.

## Current Status



Since completing the mid-project deliverable we have just finished setting up our development environment and are ready to move onto the next phase of the project being the execution phase.

# Progress for 23rd April

## Week Six

In the last week Peter has completed constructing the website and the importing data module has also just been completed. We found that the data we needed was located at hg.python.org, mail.python.org and bugs.python.org and using the nokogiri software we were able to scrape the data from those websites which will be sent to the front end system on the website when it is developed.

## Decision Table

|  |  |  |
| --- | --- | --- |
| Description | Decision | Priority |
|  |  |  |
| Website Language | Peter chose to use PHP and Javascript as the web languages as he is most comfortable with them since he is doing most of the website development.  These languages are compatible with Ruby. | High |
| Ruby | We are no longer using Ruby on Rails and have just settled with Ruby, we found that we didn’t exactly need all the tools that Rails was able to provide and for beginners of Ruby we didn’t want to overcomplicate ourselves. | Neutral |
|  |  |  |

## Issues

We found that importing the data was a more difficult task as we thought as the formats differ so easily. Also the Python repositories and archives are so broad we didn’t know where to start, we settled that the best places to get our data from would be the links listed above.

## Action Items

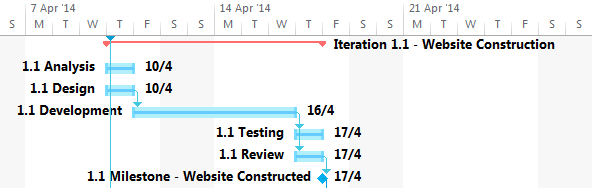
### Completed

We have completed the import data section. Peter has successfully set up the website.

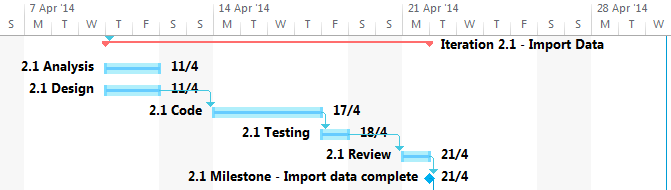
### Assigned

Now that the website has been set up and all tools installed for developing the visual data, Peter must implement a user management system with different type of users having different permissions.

## Current Status



Peter has set up hosting of the web server that will contain the Optimus software.



The importing of data module has been completed this week.

# Progress for 30th April

## Week Seven

This week was pretty light on the work for us as we only had the user management system to complete which was mainly Peters responsibility as front-end developer. We felt that it would be best if he mostly handled that so that he would be very familiar with the system for when we get up to data visualization. During this time the rest of us started early on doing some researching on data manipulation and data visualization which will be our sole focus for the next two weeks over the break.

## Decision Table

|  |  |  |
| --- | --- | --- |
| Description | Decision | Priority |
|  |  |  |
| User Permissions | We settled on three different type of users being guests, who could only read data, developers, who could read and write data and an administrator who would have greater access to the system. | Neutral |
|  |  |  |
|  |  |  |

## Issues

No issues arose for us during this week.

## Action Items

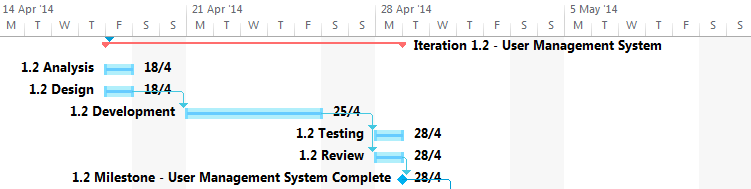
### Completed

A user management system has been successfully implemented to work with the Optimus website.

### Assigned

Although work has begun on manipulation and visualization of data, the modules for those need to be completed within two weeks in order to stay on track with our project plan. There is a short break next week however we have agreed that it would be best to continue work from home over this break to stay on top of work.

## Current Status



The user management system actually was completed in less time than shown which gave us the opportunity to begin work on the next two modules which we feel will be substantially more complex and difficult to develop and implement.

# Progress for 14th May

## Week Nine

We are currently almost finished completing the data visualization and manipulation modules. Our development team has split in order to concurrently complete these sections. These were large sections of the project and have taken two weeks to complete, including going over the short break where we have still contributed to it. In visualizing data it required creating a link between the back-end development which took in the raw python data and the front-end being the website which took in JSON files and visualized that data.

## Decision Table

|  |  |  |
| --- | --- | --- |
| Description | Decision | Priority |
|  |  |  |
| Visualising Tool | After some research we found that the best tools to use in order to visualize the python data would be google charts and JChartFX. | High |
|  |  |  |
|  |  |  |

## Issues

We found that it was tricky to merge the work of two different development teams being the front-end website and the back-end data parsing. Through sending the imported data into a local database and retrieving JSON files which will then be sent to the front-end PHP and Javascript code to handle and display to the user we were successful in this.

## Action Items

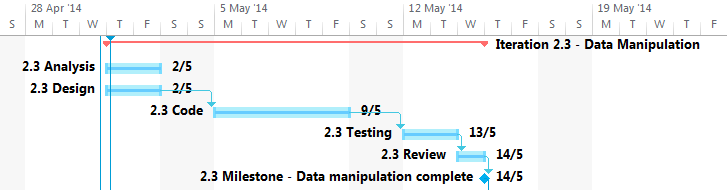
### Completed

We have done most of the visualization including setting up the user management system for the website. Some

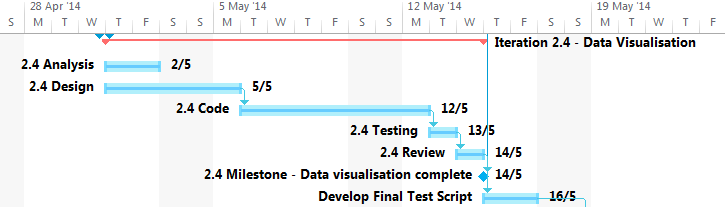
### Assigned

All members are taking part in completing the manipulation and visualization of data modules to be completed soon.

## Current Status



We are currently in the ending stages of manipulating data.



Data visualization is currently still in the middle stages of coding.

# Progress for 21st May

## Week Ten

After having our product functioning we have moved on to finalizing this project and completing documentation during the evaluation stage. Business case and effort estimation have been completed.

## Issues

We had some slight issues with finishing up with our data visualization we came to the point where we merged everybodys work together to complete the final product, in the end we manage to successfully merge systems (front/back end)

## Action Items

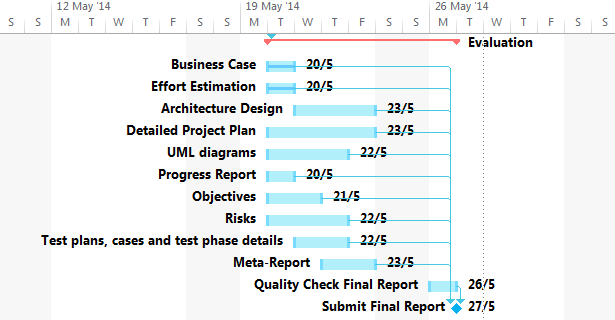
### Completed

Kurt has completed the effort estimation section while Josh completed the business cases. Last week Peter finalized the data visualization module and Jamie and James both completed the data manipulation module, thus giving us a finished product. Peter also went back and updated the website design to a more user friendly one which is greatly aesthetically pleasing.

### Assigned

Kurt has been assigned to do the UML diagrams, progress report and the meta-report. Josh has been given the task to complete the project plan, objectives and risks. Jamie is doing tests. Peter is doing the architecture design and James is doing objectives.

## Current Status



So far we have completed the business case and effort estimation sections. We still have five days until this project is due and we are on track and will be finished by then.

Meeting Records

**18th March**

Jamie, Kurt, Josh

Decided on agile development

Preferred language would be able to support a GUI

Ideas are Cocoa, C++, Java, VB

Code standards guide among the group to make developing less confusing

Action items

- Design plan and business case

- Initial SRS

- Objectives

- Measure effectiveness

- Milestones

- Make git repository

- Risk analysis

**25th March**

Looked into making it web based using the D3js tool for showing graphs/visualising the data in an aesthetic way to the user

Swinging towards using javascript as D3js tool is in javascript so it makes sense to code it in the one language.

Planning on having multiple users on program at once so MySQL > SQLlite.

Github has been setup

Work action items for people

- Feasibility Study = Jamie, Kurt

- SRS = Peter, Kurt

- Detail Plan = Willie, JB

Peter in charge of front end

Back end decide on language that is easy to learn.

Such as Ruby, PHP, JS, EngineX. They can all talk to the database through MySQL

Look at PHP and Ruby and see what we like to write in.

How many people?, Multiple = concurrent

Technical level of users, Click button do something, easy aesthetic

How often update?

Get data once, thats all we need. No new bugs etc

After we connect, no new data

Security requirements?

Needs password with logins with permissions, different levels. admin/security/user etc

**2nd April**

Notes with client in josh/peter files

Objectives

Import + Parse data (svn/bugs)

Visualise

Manipulate

Importing data

We need Ruby unarchiver (open source)

Method

Email, SVN, Bugs

Need to get Bug/Issue reports (URL)

Test cases

Execution traces/logs

Emails/discussions (Archive)

Feedback

**7th April**

Kurt

* Milestones
* Progress report

James

* Detailed plan

Josh

* Archive map

Jamie

* Feasibility Study
* Code of Conduct

Peter

* SRS

**16th April**

List of resources/versions to use in development

* Homebrew 0.9
* Sequel Pro 1.0.2
* Xcode 3.1
* Ruby 2.1.1

List of tools to use

* Nokogiri
* Ruby on Rails
* Github
* MS Office
* SQL Server

**23rd April**

Do we need archive map?

Use bugs.python.org for bug data. Can download as csv.

Hg.python.org has other data we may need to show rather than scraping straight from web documentation.

Mail is at mail.python.org. Fairly straight forward data.

Need to scrape into ruby, no rails anymore.

Make local database for each module since we split up in separate sections/groups.

Josh and James look into nokogiri as a ruby scraper to get raw data.

Need database setup to put imported data into

**30th April**

Users could be admin, guests, project manager, python developer etc

Some can commit changes or edit files where some can’t.

Need to work on updating appearance of website rather than a simple html file. Needs to appeal to client and customers. Simple design with tabs?

**14th May**

Looking at graph tools, maybe google charts, D3js, chart.js, JChartFX.

Research and test out different tools to see which is simplest. Peter get experience using graphing tools.

Need to get more front end stuff don’t to mix in with

**21st May**

Josh

* Objectives
* Detailed plan
* SRS update

Kurt

* Project progress
* Architecture
* UML diagrams
* Risks

Jamie/Peter

* Tests
* Business case
* User manual

James

* Prepare presentation
* Touch up software