Project Proposal on
Epidemic Analyzer

Biju Ale
College ID: 160026
NCC ID: 0016399
Computing Project
Level 5 Diploma in Computing
Softwarica College of IT & E-Commerce,
Kathmandu, Nepal

July 14, 2017

Table of Figures

Table of Contents

Introduction

On March 23, 2014, an epidemic of Ebola Virus Disease (EVD) in Guinnea was reported to the World Health Organization (WHO). On August 8, the situation was declared to be a “public health emergency of international concern”. (PMC NCBI, 2014) . The following 2 years saw a total case fatality of 11310 in West Africa, as the EVD diffused in Sierra Leone and Liberia. (cdc.gov, 2016)

In such milieu of a humanitarian crisis, the responders including public health personnels require epidemic data and its analysis. Such information is paramount for decision makers whose discernments and actions have direct impact on the control of the outbreak, consequently saving lives. Thus I propose to develop an epidemic analysis system that aids in the acquisition of data and subsequently provides certain analytics on it. I’ve named this system– “Epidemic Analyzer”.

Problem Identification

Owada, Kei and et al in their article published in the journal ‘Front Public Health’ (Owada et al., 2016) highlights some of the issues with data management during the Sierra Leone outbreak.

They identify that the district level data management was done in Microsoft Excel sheet in the form of line lists. These data included (but not limited to) a total number of new alerts, a total number of cases reported, and a total number of blood samples. There was difficulty in aggregating and merging data to report to the national level. For these reasons, the data never found its residence in a structured database and vital information were lost. I propose that “Epidemic Analyzer” will resolve this issue & additionally provide features discussed below.

Scope

Epidemic systems have a broad domain incorporating myriads of functionalities developed across several years. They require extensive research, learnings from field experience, and collaborations with experts in diverse fields. However, my project is a modest one and adheres to only a small subset of such vast domain. Some of the limitation of my project includes:

1. Designed to work with epidemic data only.
2. The system is a prototype and needs to be approved by the government or the WHO before actually being deployed in real scenarios.
3. Forms cannot be customized.
4. In the prototype, the geographical dataset is limited to few countries only.
5. Due to technical difficulties and time constraints, currently, it is developed using core java web components i.e. JSP & Servlets. Other J2EE components are not used comprehensively.

Features

1. Consolidated Case Investigation Forms (CIFs) for data collection. Draws from 3 different sources including the standardized form of WHO.
2. Statistical analysis of various health dynamics.
3. Graphical representation (bar graphs and charts) of summary data.
4. Representation of cases geographically on the map.
5. SIR (Susceptible, Infected, and Recovered) Modelling and simulation.
6. Database backup option.

Aims

1. To build a system that can capture epidemic data on ground zero and provide analytics for provincial and national level decision makers.
2. To contribute to the understanding of public health practitioners and humanitarian agencies about an epidemic situation through the introduction of information technologies.

Objectives

1. To provide a robust data entry form that captures a broad range of health data dynamics.
2. To provide essential analytics on epidemic data.
3. To provide a graphical representation of data for better comprehension & decision making.
4. To provide the simulation of disease spread and impact using SIR model.
5. To learn about the impact of information technology in international humanitarian interventions.
6. To apply the principles from academic learnings to the real world scenarios and garner experience in research and development.

Development Methods

Inapplicability of exclusively implementing agile development methodology

Based on the 12 principles of Agile, I provide the negative case as to the inapplicability of exclusive implementation of agile development methodology. The primary constraint discussed is the single major constraint that makes agile near to impossible to be implemented as a development methodology for my proposed project.

Primary constraint

- An uncanny nature of CP in Softwarica – The incompatibility of A Solo project with Agile's principles

  For reasons unknown, students enrolled in the NCC's L5DC module - 'Computing Project' (CP) are mandated by the college to undertake the computing project individually. Consequently, the entire phases of software development following this proposal needs to be executed unaccompanied.

  Checkpoints are set by the module leader at each phase and the deadlines for delivery of documentation marking the milestone (mapped to each phase) are also fixed prior to the submission of this proposal. Currently, all CP projects span 3 months before the final demonstration. In a nutshell, this means that CP is evidently (due to such mandates) a solo project. By design, there is absolutely zero team work.

  A derivative of the agile frameworks such as DSDM or Scrum is a collaborative enterprise. Simply looking at the semantics of how the 12 principles are stated i.e. the usage of 'we' over 'I' in every principle is enough to deduce that agile consists of a diverse team. (Beck et al., 2017) In fact, this is a team comprised of generally 7 plus or minus 2, based on Miller's Law. Thus, agile development methodology exclusively cannot be opted for this project. The keyword to be noted here is – 'exclusively'. In the later section, I provide a positive case for implementing a waterfall along with some agile principles incorporated.

Secondary constraints

- The inability for the system to be delivered incrementally in functional chunks.

  Given the scope of my project and the limitation of time of delivery (within 3 months), there are no modules in the system that can be released as independent chunks. Therefore the system is to be delivered as a whole. The principle of Agile stating that system must be released incrementally doesn't apply here. (Citation)

- Static requirements

  The strength of agile lies in change adaptability and thus highly anticipates frequent requirement updates by the client. The system requirements for this proposed project are fixed and are not subject to change at any point of systems development lifecycle.

Choosing waterfall-agile hybrid development methodology

Agile or Waterfall – Clarifying false dichotomy.

Miller et al at the Carnegie Mellon University (Palmquist et al., 2013) points out in their paper that the new world of Agile and the traditional world's waterfall based methods are not opposites; but are just different perspectives which place different emphasis on similar parts. In its simplest form, as illustrated below, the Waterfall model sets the scope up front, and allows for performance, schedule, and cost to vary. Agile sets the cost, schedule, and performance up front then allow the scope to vary.

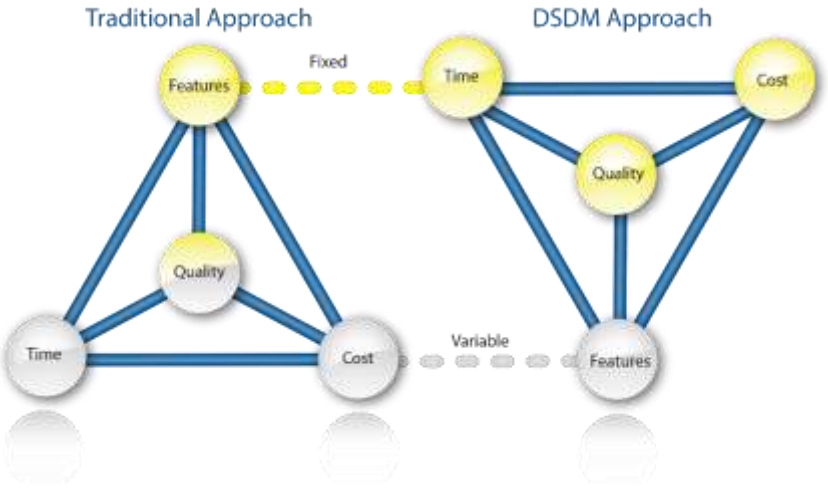

Figure 1: Iron Triangle comparison of Agile (DSDM) & Waterfall.(Source: dsdm.org, 2016)

Miller et al, further clarifies that "Depending on the project's needs, a combination of both methods may be most appropriate. It does not have to be an either-or proposition." (Palmquist et al., 2013) I base my choice of development methodology on their suggestion that development methodology may not be necessarily dichotomous in choice.

The need for a Hybrid approach

Prior to this section, I provided the negative case against agile as 'exclusive' development methodology i.e. for this project. Here, I delineate the positive case for a hybrid approach. I draw from both the new world of agile and the traditional world of the waterfall. Specific aspects incorporated of both worlds with the justifications are presented below.

Traditional Principles	Incorporated in the waterfall-agile hybrid approach?	Justification
Plan the work—especially the budget, schedule, and deliverables—to the maximum extent possible before beginning any design or code.	Yes	Project requirements are static as stated in the scope section of this proposal.
Lock down requirements to prevent scope creep.	Yes	Since milestones and deadlines for each SDLC phases are fixed, scope creep is a high-risk factor but prevented with waterfall.
Institute multiple reviews to provide	Yes	The project is supervised by the current

senior leadership oversight as well as to serve as gates for continued work		module leader – Sudeep Bajimaya. He has set the milestones and several deadlines of deliverables Hence, this principle is included.
Capture all details with extensive documentation	Yes	The deliverables' deadline set by the module leader for each phase necessitates the documentation be thorough.

Figure 2: Principles incorporated from Waterfall Model. (These principles were taken from Miller et al's paper - (Palmquist et al., 2013)

Agile Principles	Incorporated in the waterfall-agile hybrid approach?	Justification
Our highest priority is to satisfy the customer through the early and continuous delivery of valuable software.	No	This is an academic project with no current ties to a real client. The target beneficiaries are assumed for future.
Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.	No	Project requirements are static as stated in the scope section of this proposal.
Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.	Yes (Partially)	The project could be modularized but modules may have dependencies to be fully functional. Modules can be developed and tested parallel.
Business people and developers must work together daily throughout the project.	No	There are no business people and developers. This is s a solo project.
Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.	No	This is a solo project.
The most efficient and effective method of conveying information to and within a development team is a face-to-face conversation.	No	There is no development team. This is a solo project.
Working software is the primary measure of progress.	Yes (Partially)	The project is delivered as a whole once at the end of SDLC phase.
Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.	Yes	Since this is a solo project all stakeholders are assumed for future correspondence once the software is developed. The project is developed through informed assumptions and supervision of module leader.
Continuous attention to technical excellence and good design enhances agility.	Yes	This is an academic project and not a commercial one. Hence, the focus is on the quality of system through implementation of

		learnings in the past modules such as ADI and ISA.
Simplicity--the art of maximizing the amount of work not done--is essential.	Yes	
The best architectures, requirements, and designs emerge from self-organizing teams.	No	There is no team. This is a solo project.
At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.	No	There is no team. This is a solo project.

Figure 3: Principles incorporated from Agile Model. (These principles were taken from the Agile Manifesto's web page- (Beck et al., 2017)

Design Pattern

MVC

Model View Controller (MVC) is used to manage the architectural complexity of the project. Domain classes obtained from NLA & ER diagram are translated to Java class files. These are placed in the model package. The view consists of all GUI components including data entry forms and frames and panels to display charts and statistical analysis. The controller acts as a façade and wrapper for core logics of the system. It acts as a bridge between model and view.

DAO

All models likely to persist in the database are handled by DAO classes. It will contain all DML logics and object-data relational mapping implementaions.

Simple Factory

Connection Factory – for JDBC connection pre-requisites. All DAO classes know about this Connection Factory

Model & GUI Factory – used in conjunction with singleton if necessary. This can be used to separate object instantiation logic.

Singleton

This is highly likely to be used in my project. It will provide global access to classes operations while limiting its creation to single instance throughout the JVM session.

Observer

It may be used in the project where multiple observers are dependent on the common subject and subject needs to push (or observer needs to pull from the subject) updated data to observers. It will most likely be used to generate graphs and charts.

Tools

IDE	Netbeans 8.1 & Eclipse 4.7 (Oxygen)
Programming Language	Java 8 (Runtime version: 1.8.0_131-b11) J2EE (JSP, Servlet)
Framework (proposed as tentatively only.)	Hibernate ORM 5.2.10 SpringBoot or equivalent
Programming Paradigm	Object Oriented
Database	MariaDB 10.2.7 (formerly MySQL)
Server solution stack	XAMPP Control Panel v3.2.2
External Java Libraries (for generating graphs & charts)	JFreeChart, charts4j, GRAL
Modelling tool (ER, Class diagram, sequence, activity, use case)	Visual Paradigm CE 14.1

Project Plan

Work Breakdown Structure (WBS)

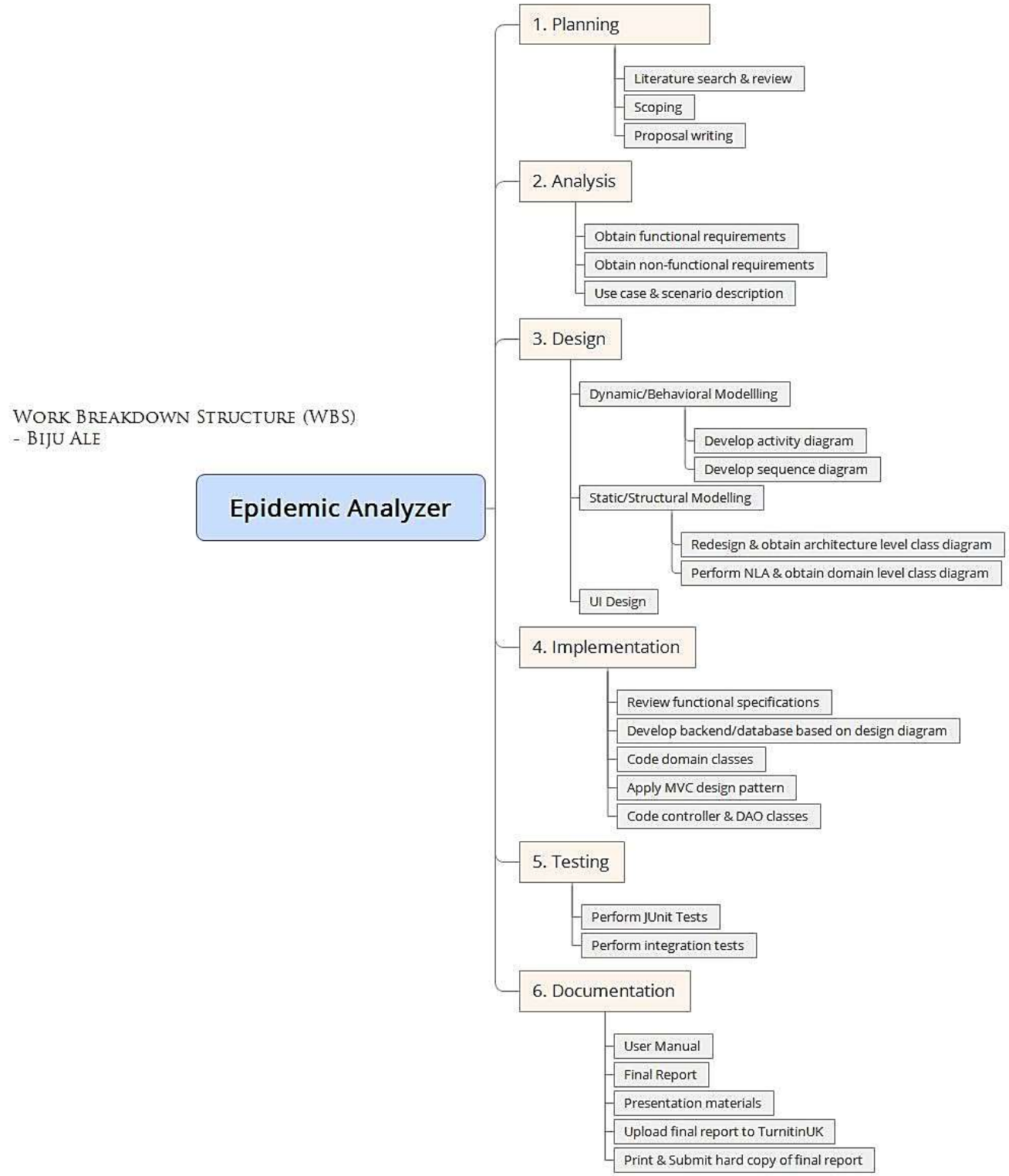

Figure 4: Project WBS

WBS#	Task Name	Time estimates (Days)
0.	**Project - Epidemic Analyzer**	**126**
1.	**Planning**	18
1.1	Literature search & review	8
1.2	Scoping	4
1.3	Proposal writing	6
2.	**Analysis**	26
2.1	Obtain functional requirements	10
2.2	Obtain non-functional requrements	10
2.3	User case & scenario description	6
3	**Design**	13
3.1	Static/Structural Modelling	6
3.1.1	Perform NLA & obtain domain level class diagram	2
3.1.2	Redesign & obtain architecture level class diagram	2
3.1.3	Obtain ER diagram	2
3.2	Dynamic/Behavioral Modelling	6
3.2.1	Develop sequence diagram	2
3.2.2	Develop activity diagram	3
3.3	UI design	1
4.	**Implementation**	**26**
4.1	Review functional specifications	4
4.2	Develop backend/database based on design diagram	3
4.2	Code domain classes	5
4.3	Apply MVC pattern	5
4.4	Code controller & DAO classes	10
5.	**Testing**	**22**
5.1	Perform Junit Testing	15
5.2	Perform Integration testing	7
6.	**Documentation**	**15**
6.1	User Manual	5
6.2	Final Report	6
6.3	Presentation materials	2
6.4	Upload final report to TurnitnUK	1
6.5	Print & submit hard copy	1
7.	**Demonstration**	**1**
7.1	Demo & viba	1

Figure 5: Project WBS tabulated with Time Estimate

Work Breakdown Structure aims to provide the actual work done in the project at various level. The broad areas of work are shown at higher levels where as the exact activities are seen in the lower level break downs of respective work. The time estimates are approximated and are subject to change due to unforeseen situations in future.

Milestones

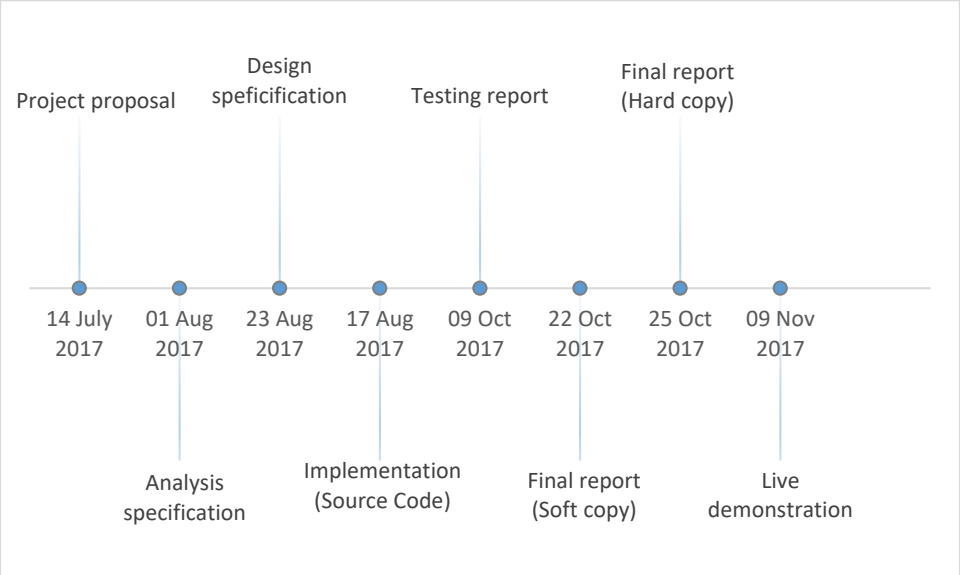

Figure 6: Project's major milestones

There are 8 major milestones for this project. The milestones generally maps to each phases of the SDLC cycle. They are beiflly discussed below:

1. Project Proposal - 14th July 2017

This marks the milestone for planning which involves literature search & review, scoping. Since these are pre-requisite to the proposal, the final proposal would state that the planning phase is complete.

2. Analysis Specifications -1st August 2017

After the proposal is submitted and approved by the supervisor, the next series of activities include a high level analysis of the system that draws from ISA analysis methdologies and then moves onto the static and dynamic analysis of the system.

Once all the activities of anlysis are presented in a structured report, this report would mark the milestone for the analysis phase of SDLC.

3. Design Specification -23 Aug 2017

Once the analysis is executed, the next series of activities would revolve around 'proof of concept'. This would involve obtaining the detailed architecture of the system. Moreover, this stage would model the dynamic and static aspects of the system. Hence, once the design specification delineates all these elements, it can be considered a milestone.

4. Implementation - 17 Aug 2017

Arrival at this milestone means that we are ready to translate our implementation independent design into actual codes in our preferred choice of language, in my case - Java. Thus the domain level design of classes are first layed out in the MVC pattern & additional classes incorporated in architecure design is included in the code base. Finally, as the code base is readied, this would mark the milestone for a implementation.

5. Testing report - 09 Oct 2017

Testing report marks a crucial milestone which suggests that the high level analysis have been implemented in a programming language. And that the unit and integration testing has been carried out. The testing report signifies that the system is both verified and validated.

6, Final report (soft copy) - 22 Oct 2017

Once all the prior milestones are crossed, the next stage is a matter of consolidating all our analysis, design, & implementation into a single documentation that is submitted. This will also include user help specifications. The document once uploaded to TurnitinUK for plagiarism inspection will mark the milestone for final report.

7. Final report (hard copy) - 25 Oct 2017

This is the semi-final milestone before the completion of the entire CP project. This invloves hard copy presnetation for the project to the module supervisor.

8. Live demonstration

This is the final milestone. Reaching this means that all presentation materials are prepared and the projet is demonstrated among a panelist of experts and faculties for the final marking of project.

Schedule

[Next Page]

	Name	Duration	Start	Finish	Predecessors
1	**Epidemic Analyzer**	**137 days?**	**6/26/17 8:00 AM**	**11/9/17 5:00 PM**	
2	**Planning**	**18 days**	**6/26/17 8:00 AM**	**7/14/17 8:00 AM**	
3	Literature search & review	8 days	6/26/17 8:00 AM	7/3/17 5:00 PM	
4	Scoping	4 days	7/4/17 8:00 AM	7/7/17 5:00 PM	3
5	Proposal writing	6 days	7/8/17 8:00 AM	7/13/17 5:00 PM	4
6	Planning complete	0 days	7/14/17 8:00 AM	7/14/17 8:00 AM	5
7	**Analysis**	**26 days**	**7/15/17 8:00 AM**	**8/9/17 5:00 PM**	**6**
8	Obtain functional requirements	10 days	7/15/17 8:00 AM	7/24/17 5:00 PM	
9	Obtain non-functional requirements	10 days	7/25/17 8:00 AM	8/3/17 5:00 PM	8
10	Use case & scenario description	6 days	8/4/17 8:00 AM	8/9/17 5:00 PM	9
11	Analysis complete	0 days	8/9/17 5:00 PM	8/9/17 5:00 PM	10
12	**Design**	**13 days?**	**8/10/17 8:00 AM**	**8/23/17 8:00 AM**	**11**
13	**Static/Structural Modelling**	**6 days**	**8/10/17 8:00 AM**	**8/15/17 5:00 PM**	
14	Perform NLA & bbtain domain level class diagram	2 days	8/10/17 8:00 AM	8/11/17 5:00 PM	
15	Redesign & obtain architecture level class diagram	2 days	8/12/17 8:00 AM	8/13/17 5:00 PM	14
16	Obtain ER diagram	2 days	8/14/17 8:00 AM	8/15/17 5:00 PM	15
17	**Dynamic/Behavioral Modelling**	**6 days?**	**8/16/17 8:00 AM**	**8/21/17 5:00 PM**	**16**
18	Develop sequence diagram	3 days?	8/16/17 8:00 AM	8/18/17 5:00 PM	
19	Develop activity diagram	3 days?	8/19/17 8:00 AM	8/21/17 5:00 PM	18
20	UI Design	1 day?	8/22/17 8:00 AM	8/22/17 5:00 PM	19
21	Design complete	0 days	8/23/17 8:00 AM	8/23/17 8:00 AM	20
22	**Implementation**	**26 days**	**8/23/17 8:00 AM**	**9/17/17 5:00 PM**	**21**
23	Review functional specifications	4 days	8/23/17 8:00 AM	8/26/17 5:00 PM	
24	Develop backend/database based on design diagram	2 days	8/27/17 8:00 AM	8/28/17 5:00 PM	23
25	Write code for domain classes	5 days	8/29/17 8:00 AM	9/2/17 5:00 PM	24
26	Apply MVC pattern	5 days	9/3/17 8:00 AM	9/7/17 5:00 PM	25
27	Write code for controller & DAO classes	10 days	9/8/17 8:00 AM	9/17/17 5:00 PM	26
28	Implementation complete	0 days	9/17/17 5:00 PM	9/17/17 5:00 PM	27
29	**Testing**	**22 days**	**9/18/17 8:00 AM**	**10/9/17 5:00 PM**	**28**
30	Perform Junit Testing	15 days	9/18/17 8:00 AM	10/2/17 5:00 PM	
31	Perform integration testing	7 days	10/3/17 8:00 AM	10/9/17 5:00 PM	30
32	Testing complete	0 days	10/9/17 5:00 PM	10/9/17 5:00 PM	31
33	**Documentation**	**15 days**	**10/10/17 8:00 AM**	**10/25/17 8:00 A...**	
34	User Manual	5 days	10/10/17 8:00 AM	10/14/17 5:00 PM	
35	Final report	6 days	10/15/17 8:00 AM	10/20/17 5:00 PM	34
36	Presentation materials	2 days	10/21/17 8:00 AM	10/22/17 5:00 PM	35
37	Final report upload to turnitin	1 day	10/23/17 8:00 AM	10/23/17 5:00 PM	36
38	Print and submit hard copy	1 day	10/24/17 8:00 AM	10/24/17 5:00 PM	37
39	Documentation complete	0 days	10/25/17 8:00 AM	10/25/17 8:00 AM	38
40	**Demonstration**	**1 day?**	**11/9/17 8:00 AM**	**11/9/17 5:00 PM**	
41	Demo & Viba	1 day?	11/9/17 8:00 AM	11/9/17 5:00 PM	39
42	Demonstration complete	0 days	11/9/17 5:00 PM	11/9/17 5:00 PM	41

Figure 7: Project Schedule

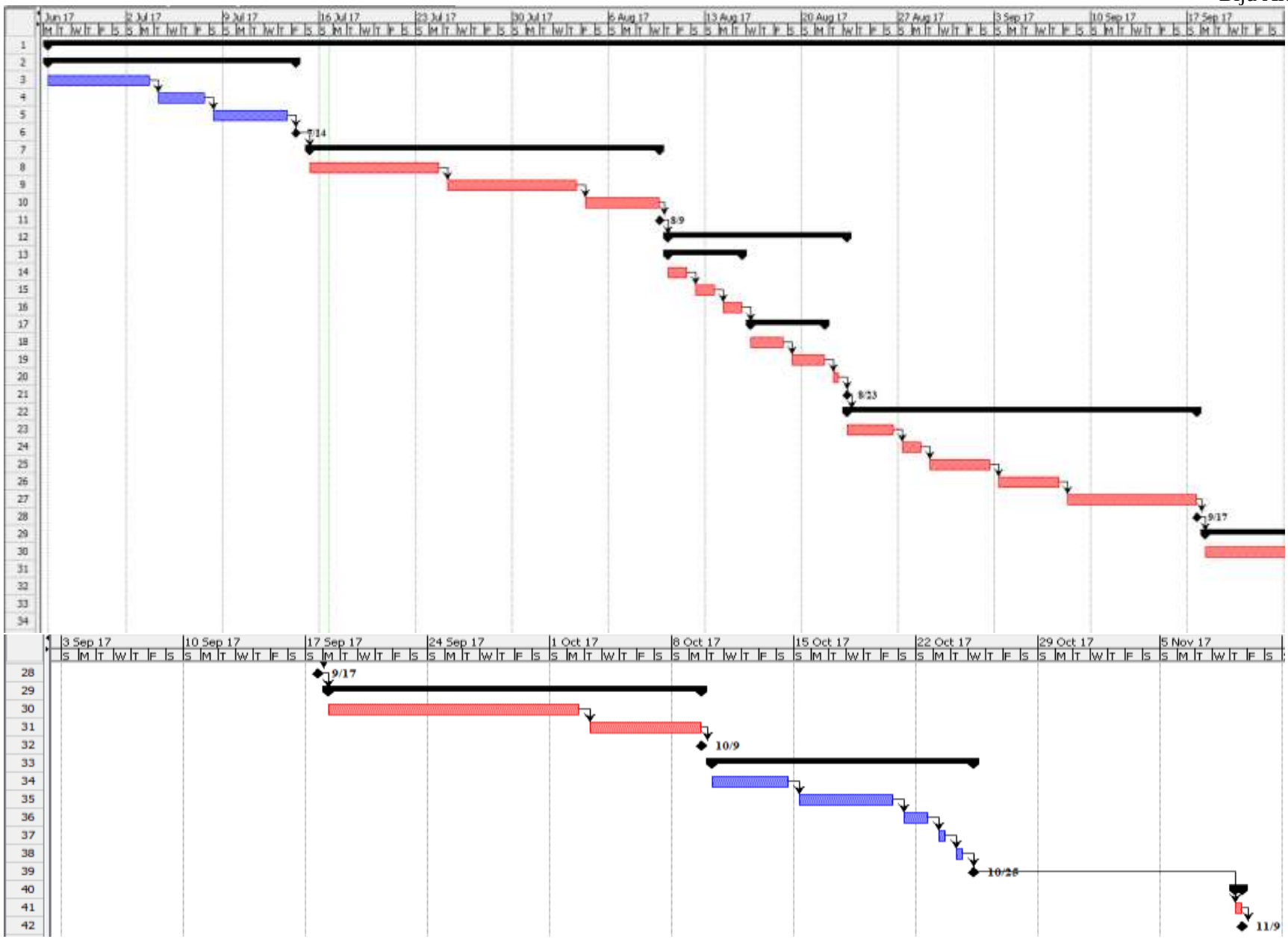

Figure 8: Gantt Chart (Split in 2 for full view)

Risk Management

The universe looms with ubiquitous uncertainties and the unwanted event can interrupt the flow and execution of any other events at any instance. The solution is to be best prepared and anticipate potential risks early on while having some kind of plan to mitigate or navigate through such situations. This is the philosophy of risk management.

Just as project management involves the development of a project plan and control of the project using that plan as the project progresses, risk management involves the identification of risk. Risk management involves identification of risks at the project's outset and the control of those risks as the project unfolds. (Dawson, 2015)

Four steps to risk management are:

1. Identify risks
2. Access impact of risk
3. Alleviate critical risk
4. Control risk

Likelihood	**Value**
Low	1
Medium	2
High	3

Figure 9: Risk Likelihood values (Turner, 1999)

Consequence	Value
Very Low	1
Low	2
Medium	3
High	4
Very High	5

Figure 10: Risk consequence values (Turner, 1999)

Taking reference from the above tables, to estimate the impact of each identified risks we use,
Impact = Likelihood × Consequence

Based on Turner's (Turner, 1999) quantitative measure for assessing risk, & based on the guidelines for conducting such assessment provided by Dawson (Dawson, 2015), I've presented below, the risk management matrix for my project.

Risk Type	Risk	Likeli-hood	Consquence	Impact	Action Type	Action
Non technical	Scope Creep – Uncontrolled & incremental growth of project's scope at any point of project causing cost or time overrun. (Knight, Thomas, and Angus, 2013)	1	5	5	Avoidance	Follow requirements specifications from analysis phase. Remind oneself of impending deadlines using reminder applications and wall calendars.
	Gold Plating – It refers to intentionally adding extra features or functions to the products which were not included in the scope statement. (Usmani, n.d.)	1	4	4	Avoidance	Follow requirements specifications from analysis phase. Remind oneself of impending deadlines reminder applications and wall calendars.
	Natural Disaster	1	5	5	Contingency	Regularly backup entire project directory in Google drive.
	Deadline overruns	2	5	10	Contingency	Remind oneself of impending deadlines reminder applications and wall calendars. Commit to earlier deadline over actual stipulated deadlines.
Technical	Hardware failure	2	4	8	Contingency & avoidance	Perform system diagnosis and hardware condition checks on the local machine at the service center. Apply upgrades and fixes if necessary. Use power surge protection. Regularly backup project to google drive. Keep a backup machine in the vicinity on standby.
	Hard disk failure	1	5	5	Contingency &	Clone the local machine's hard drive

					avoidance	once before initiating the project. Regularly backup project on google drive.
	Technical difficulties (ignorance of algorithm, sticky bugs, problems with modeling) during design and implementation.	3	4	12	Deflection	Get expert advice from supervisor and faculty members. Research on the web and contact industry experts if required.
	Steep algorithmic complexity to achieve certain functionality.	3	4	12	Deflection	Get expert advice from supervisor and faculty members. Research on the web for APIs, and libraries to perform such tasks.

Configuration Management

All project artifacts reside in two locations – the local machine, where all the files are created and the GitHub repository (https://github.com/BijuAle/) under the main folder titled – CP Project. Both locations are in sync using the Git Bash.

The directory is based on major phases of SDLC. Each phase has a folder and all documents, code base, and modeling diagrams reside in respective folders. The 'Project Management' folder contains all the research materials and planning documents such as research literature, proposal, and schedules. There is a folder 'Backups' where the entire project is backed up and saved in sub-folder named by date and time of the time of backup. This will help me rollback changes made at particular instances of significant commit points. Finally, the whole project structure is backed up in Google Drive regularly.

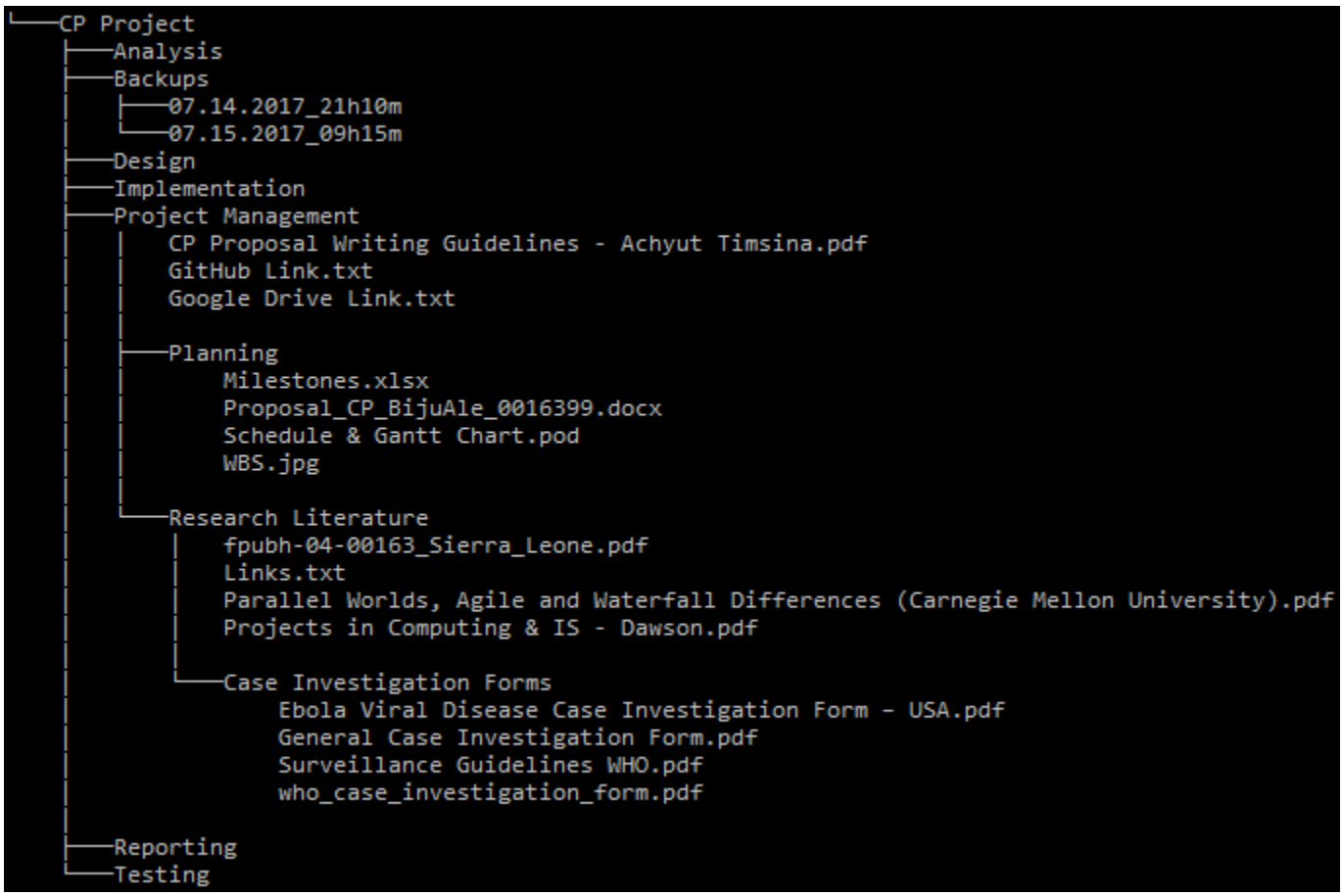

Figure 11: Directory Structure for Project in the local machine.

Conclusion

This document has sought to propose ‘Epidemic Analyzer’, an epidemic data analysis software that aids in the acquisition of health data in humanitarian situations and provides a level analytics and data representation. I began the proposal by first setting up a scenario for the domain of my system. Citing a paper on data management issue at district level during the 2014 Ebola outbreak, I identified the problem statement. Then, the greater aim was set which revolved around contributing to the humanitarian responders by leveraging of information system. Moving on to define the scope of my project, I clarify the limits of my projects. In the development methodology section, I provide arguments for opting a hybrid waterfall-agile methodology specially for this project. The time and the solo nature of the project were discussed and salient features of both methodologies were drawn to form the hybrid method. The entire project was planned across 3 months and details schedule, WBS, and milestones were illustrated. Then, a risk management matrix for my project was designed along with mitigation strategies. Finally, configuration management was discussed briefly outlining a simple yet effective folder structure for the project based on SDLC phases.

References

PMC NCBI. (2014). *Ebola Virus Disease in West Africa — The First 9 Months of the Epidemic and Forward Projections*. [online] Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4235004/ [Accessed 16 Jun. 2017].

cdc.gov. (2016). *2014 Ebola Outbreak in West Africa - Case Counts | Ebola Hemorrhagic Fever | CDC*. [online] Available at: https://www.cdc.gov/vhf/ebola/outbreaks/2014-west-africa/case-counts.html [Accessed 18 Jun. 2017].

Owada, K., Eckmanns, T., Kamara, K. and Olu, O. (2016). *Epidemiological Data Management during an Outbreak of Ebola Virus Disease: Key Issues and Observations from Sierra Leone*. [online] PMC NCBI. Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4976087/ [Accessed 19 Jun. 2017].

Beck, K., Beedle, M., van Bennekum, A., Cockbyrn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., C. Martin, R., Mellor, S., Schwaber, K., Sutherland, J. and Thomas, D. (2017). *Principles behind the Agile Manifesto*. [online] Agilemanifesto.org. Available at: http://agilemanifesto.org/principles.html [Accessed 23 Jun. 2017].

Palmquist, M., Lapham, M., Miller, S., Chick, T. and Ozkaya, I. (2013). Parallel Worlds: Agile and Waterfall Differences and Similarities. *Research Showcase @ CMU, Software Engineering Institute*, CMU/SEI-2013-TN-021.

Turner, J. (1999). *The handbook of project-based management*. London: McGraw-Hill.

Dawson, C. (2015). *Projects in computing and information systems*. Harlow: Pearson.

Knight, J., Thomas, R. and Angus, B. (2013). *Battle Scope Creep in Your Projects*. [online] Harvard Business Review. Available at: https://hbr.org/2013/06/battling-scope-creep-in-your-p [Accessed 31 Jun. 2017].

Usmani, F. (n.d.). *Scope Creep and Gold Plating in Project Management*. [online] PM Study Circle - A PMP Exam Preparation Blog. Available at: https://pmstudycircle.com/2012/09/scope-creep-and-gold-plating-in-project-management/ [Accessed 1 Jul. 2017].