Date: Jan 31, 2016

### Aparna Pavithran

### Sriharshareddy Munjuluru

### Amy Kiernan

### Sangwoo Moon

### Juan Liu

### Longhao Gao



 Moon Cake Team

Dear Diary Project Proposal

# 

INDEX

1. Summary ……………………………………………………………………………………………………….. 2
2. Why this project …………………………………………………………………………………………….. 3

2.1) Need for Dear Dairy ………………………………………………………………………………… 3

2.2) Detailed problem description ………………………………………………………………….. 3

1. Objective ……………………………………………………………………………………………………….. 3

3.1) Scope of work …………………………………………………………………………………………. 3

3.2) Design specification ………………………………………………………………………………… 4

3.3) Critical design issues, constraints and limitations ……………………………………. 4

1. Approach ………………………………………………………………………………………………………… 4
2. Project Management ……………………………………………………………………………………… 8

5.1) Iteration 1 ……………………………………………………………………………………………….. 8

5.2) Iteration 2 ……………………………………………………………………………………………….. 8

5.3) Iteration 3 ……………………………………………………………………………………………….. 9

1. Deliverables for each Iteration ……………………………………………………………………… 10

6.1) Iteration 1 ……………………………………………………………………………………………… 10

6.2) Iteration 2 ……………………………………………………………………………………………… 10

6.3) Iteration 3 ……………………………………………………………………………………………… 11

1. Team qualifications ………………………………………………………………………………………. 11
2. Appendix ……………………………………………………………………………………………………… 12

## 1. SUMMARY

Dear Diary is a web application which replicates a real life diary or personal information. Dear Diary can be used to assess day-to-day work or to keep track of work for the upcoming days. It can be also used as a reminder and also as storage of sweet memories.

Some additional features that this project fulfills include the ability to keep track of birthdays and other events, like anniversaries or meetings. Also, this application will allow the user to write notes every day, as well as save pictures and link those pictures to text, and both features can be referred to at any time. This application will provide with the functionality to save the individual data to the local system.

## 2. Why This PROJECT?

#### 2.1 Need for Dear Diary

As the software industry has grown significantly and ease of access to the Internet has increased as well, it has almost been made a necessity to convert everyday human activity into an application. A personal diary is one such functionality to automate, as days have become eventful and capturing those events in a tangible book is not always possible. A web application where saving everyday photos, writing any minute thought, and receiving appropriate notifications is of great need at this time.

There are few web applications which touch these concepts but we want to make a new design of the system where the features helps the user to access the system easily and efficiently.

#### 2.2 Detailed Problem Description

Pivotal Actors of the system:

* Admin
* Users

Admin: Admin is responsible for making changes to the website, giving limited/authorized permission to users, checking if disk space is available or not for storage, authorizing new packages for the website before making changes, sending mail to all users when system is down, and approving authenticated users to access the system.

Users: Users can register to the web site and store their personal information in their personal diary. Users have all the permissions to add or make changes to their personal diary. However, they cannot view or make changes to others system.

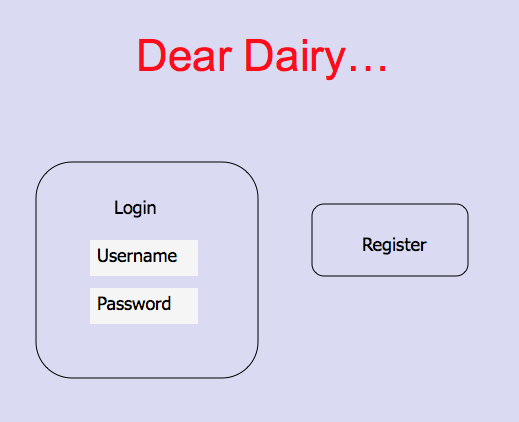
## 3. Objective

The objective of the project is to build a web application to provide a personal space to users and allow access to their system from any where around the world.

3.1. Scope of work: The system should provide a home screen with group of icons. The icons, which are as follows:



3.1.1. Register



3.1.2. Login Page

* Add Event
* Birthdays.
* Events
* Add Pictures
* View pictures.
* Add notes
* View Notes

3.1.2.1. Add Event: This is used to add events, which is used to show the information in Birthdays and the events appropriately.

3.1.2.2. Birthday button: Birthday is used to show the birthday information along with the date and name of the person, which we gave while adding event.

3.1.2.3. Events button: This is used to store other events like Anniversary, Ceremonies and other important life events.

3.1.2.4. Add Pictures: This is a button to store the pictures. This functionality helps us to upload the images.

3.1.2.5. View Pictures: The uploaded images will be shown through this button.

3.1.2.6. Add Notes: This is used to add notes by the user and is saved with date and time. User will be provided functionality to add custom name while adding these notes.

3.2. Design Specifications:

3.2.1. Functional Requirements:

1. Allow users to register to the system.
2. Allow users to login and personalize their data and store.
3. Allow users to access their data from all around the world.

3.2.2. Non Functional Requirements:

1. The project should work on windows new platforms.
2. It should be compatible with major browsers like Internet Explorer, Mozilla and Chrome.
3. Retrieval of the information should be less than 30secs.
4. Saving the information should be less than 30 secs.

3.3. Critical design issues, limitations and constraints: We see that all the events, notes and pictures are separate classes, which are underlying inside the main project class which is Diary. Coordinating between these events requires a design pattern, which is a critical design decision. Also since a diary could have many functionalities like linking the information in your diary to someone else diary which requires most sophisticated design we are limited currently to diary accessed by oneself.

## 4. APPROACH

**Purpose:** This project is a repository of a user’s list of events, photos, and notes.

**Activities:** A web application called Dear Diary is being proposed. This project involves the users uploading their events, photos and notes.

**Output:** The output will provide a web interface where user is able to upload their information and will be able to retrieve the information later. The final solution will be in the form of an executable assembly, which could easily be deployed.

**Anticipated problems:** A user wants to upload a picture with a name which is already present in the storage location (Some other user might have uploaded with same name). This could cause an issue for the user to upload or else it may replace the picture of some other user. The same instance could happen when a user uploads his personal notes. Sometimes users may accidentally retrieve the notes, which may not belong to them.

**Design Solution:**

1. Generate a separate folder for each person to store his or her individual pictures and notes. This separate folder will be automatically created in the directory when the user registers in the application.
2. Segregate the notes of the users separately to help us to divide the concern of different sections.
3. When a user uploads a picture it will be uploaded into his/her folder rather than a common one for all.
4. Similarly when the user uploads their notes it will be saved in their folder only.
5. This avoids the user from replacing the picture of some other user.
6. Also a message will be shown to the user that a picture with the same name already exists if he is replacing his own picture. Only after approval of the user it will replace the picture.

**Performance of the solution:**

1. When all the user’s information is stored at a common location – every time a user requests some data system should go through the information of all the other users.
2. This issue is resolved by our solution. When a user logs into the system she will have access to only her folder.
3. It reduces a lot of scanning logic in the system, as the comparison of information files will be less.

**Alternate solution:**

1. Map each and every file with the user’s information and store it in database with a unique ID in a table.
2. When the user requests a file he will have access to the files, which are created by them.
3. It is considered as second best solution as there will be no data clashes in this approach. But since it requires the database, there will be additional overhead of storing the user’s information in the table to map.
4. Also, retrieving will require scanning the table and then going to the file to display to the screen.

The task structure break down of this project contains four main steps.

* Gather the requirements
* Design and analysis
* Implementation
* Test

Each step can further break down into concrete tasks, which can be solved. For generating the solution concepts, there are two ways: either get a rough picture of what the solution should look like from the requirements gathering process or compare with similar products which have already been commercialized, such as Evernote.

For analyzing the performance of the solution, it is necessary to compare the solution to the requirements and also find volunteers to use the solution to give more comments to help improve the proposed system.

One alternative can be the online noting systems, such as Evernote. Because the main functionality can be implemented using similar techniques, a switch to the alternate choice can be done seamlessly if the first proposal has severe problems.

Steps for each iteration:

1. Analyze requirements.

* Iteration 1: Thoroughly go through the project and address, then document, all requirements for the scope of this project. This initial step is vital to ensuring an effective project analysis that can be built upon.
* Iteration 2: Review all requirements, and add any that have been found to be necessary after time has passed and the next steps in development have been addressed.
* Iteration 3: Once again review all requirements and ensure that nothing has been looked over, and add any requirements that came up once development of the application has progressed.

2. Develop use cases.

* Iteration 1: Once the requirements are written out as a detailed list, the next step is to write out use case texts for each requirement. Correct identification of the actor, system, and all steps that are involved in meeting each requirement is a must.
* Iteration 2: Refine use case texts written in Iteration 1, addressing any issues that may have come up in further review and implementation of the use cases.
* Iteration 3: Use cases ideally should not be changed much in this iteration, though some minor modifications may be necessary to ensure quality.

3. Feature submission for the current iteration.

* Iteration 1: The features in this iteration will include the initial web page with a visual of all the buttons, functionality to add events, and the ability to view notes.
* Iteration 2: The features in this iteration will include functionality to save notes and add pictures.
* Iteration 3: The features in this iteration will include functionality to save birthday lists as well as to add text to already added pictures.

4. Design.

* Iteration 1: Focus on setting up the home screen of the web page with all necessary buttons. Two other pages that will need design focus in this iteration is the one for adding events, and viewing notes.
* Iteration 2: It will be necessary to design the web pages for adding notes and pictures. Design focus needs to be directed to how the functionalities of both adding and viewing notes will be displayed to the user.
* Iteration 3: Address the design of adding text to pictures and how that will be displayed to the user.

5. Make modifications to the current system

* Iteration 1: Not applicable in this iteration, as the current system is not yet developed.
* Iteration 2: Before moving onto implementation of the new features, ensure that the initial system is working as desired with no issues.
* Iteration 3: Before moving onto implementation of the new features, ensure that the initial system is working as desired with no issues.

6. Implement solutions.

* Iteration 1: Divide up tasks of developing the initial web page, the adding events web page, and the note viewing web page among group members.
* Iteration 2: Divide up tasks of developing the web page for adding and viewing notes, as well as the web pages for adding and viewing pictures.
* Iteration 3: Develop the functionality to add text to already added pictures.

7. Implement testing for current solutions.

* Iteration 1: Assign group members to test each web page being developed in this iteration, ensuring that all promised functionality is working as expected.
* Iteration 2: Assign group members to test each web page being developed in this iteration, ensuring that all promised functionality is working as expected. Testers also need to ensure that user experience is enjoyable across all parts of website.
* Iteration 3: Assign group members to test each web page being developed in this iteration, ensuring that all promised functionality is working as expected. Testers also need to ensure that user experience is enjoyable across all parts of website.

8. Code submission.

* Iteration 1: Make sure all code for the features addressed in this iteration is merged together with all group members’ work and produced as a single code submission.
* Iteration 2: Make sure all code for the features addressed in this iteration is merged together with all group members’ work and produced as a single code submission. Also make sure this iteration’s code is merged with previous iteration’s code.
* Iteration 3: Make sure all code for the features addressed in this iteration is merged together with all group members’ work and produced as a single code submission. Also make sure this iteration’s code is merged with previous iteration’s code.

9. Merge new version with the previous iteration.

* Iteration 1: Not necessary for the first iteration.
* Iteration 2: Ensure that features added in this iteration are combined smoothly with the features from Iteration 1. An example would be to make sure that adding and viewing notes is addressed in an efficient way for the user to interact with.
* Iteration 3: Ensure that features added in this iteration are combined smoothly with the features from Iteration 2.

## 5. Project Management

5.1. Iteration 1:

Milestone - 1:

* Task 1: Requirement Analysis- This includes gathering requirements specific to the web application. Build the use case diagrams and use case text during this phase.
* Task 2: Designing the system: Start designing the system parallel with the Analysis. This phase can expect sequence diagrams, a process model, and a class diagram.

Milestone - 2:

* Task 1: Developing the web page to enter the events. The data should be stored in a location as per the design.
* Task 2: Developing the web page to display the buttons. Clicking on a button should redirect to appropriate page.
* Task 3: Displaying the web pages related to birthday, events and notes.

5.2. Iteration 2:

Milestone 1:

Task 1: Reviewing the design and identifying the required design patterns.

Task 2: Analyzing the new requirements like adding the pictures. Save and display the notes.

Task 3: Modifying the existing design to allow the new requirements addition.

Milestone 2:

Task 1: Developing the functionality to save the notes and displaying it as planned in the design phase.

Task 2: Developing the functionality to add the pictures. This includes uploading the pictures and displaying the uploaded pictures on screen.

Task 3: Testing.

5.3. Iteration 3:

Milestone 1:

Task 1: Developing a functionality to save the birthday list which we have created to the local system. (Download).

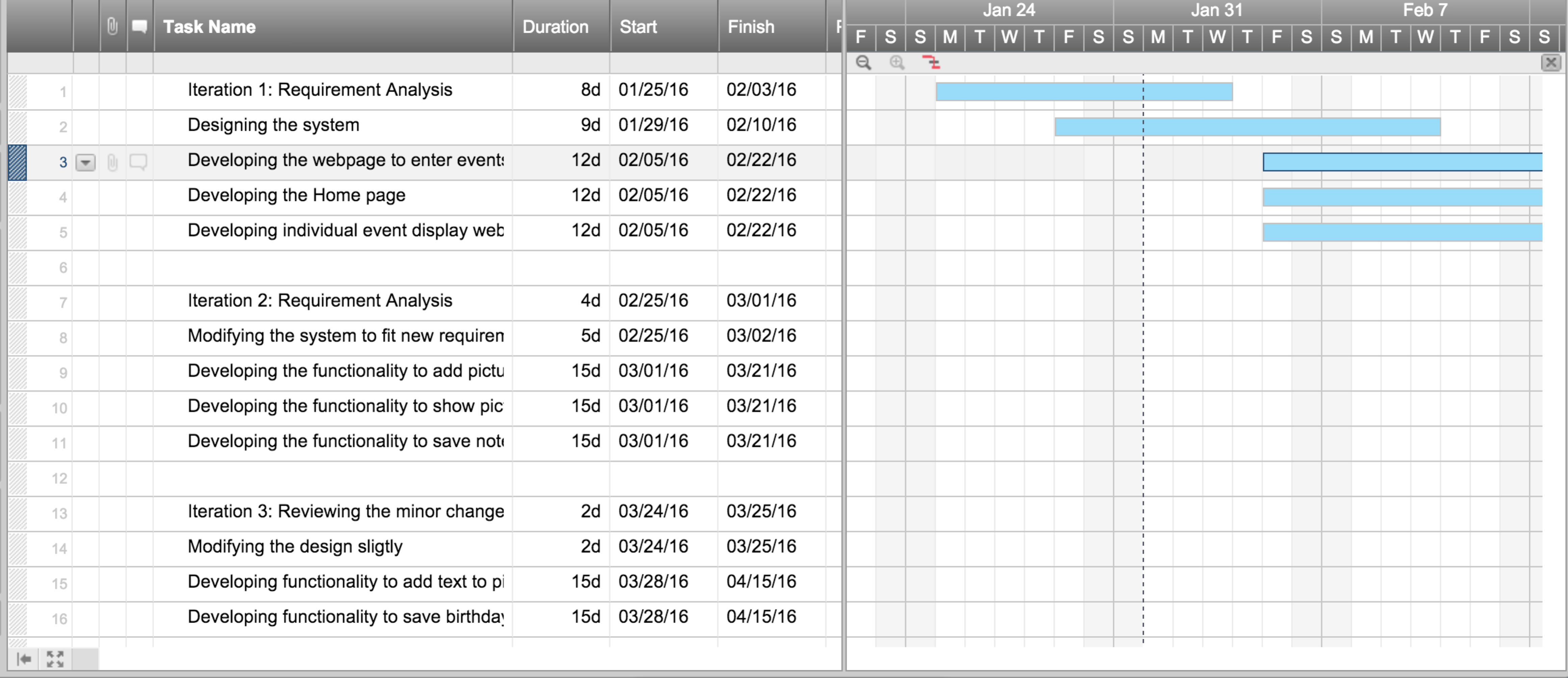
Task 2: Entering the pictures with some text. This helps to relate the picture with some description about it.

Milestone 2: Regression testing of the application and prepare the demo for the application.

**Resource scheduling table:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Names** | **Milestone 1** | **Expected Completion 1** | **Status** | **Milestone 2** | **Expected Completion 2** | **Status** |
| Aparna | Requirement Analysis | Feb-3-2016 | Started | Task 1 & Task 3 | Feb-22-2015 |  |
| SriHarsha | Design | Feb-5-2016 | Started | Task 2 & Task 3 | Feb-22-2015 |  |
| Amy | Requirement Analysis | Feb-3-2016 | Started | Task 2 & preparing for the presentation. | Feb-22-2015 |  |
| Moon | Requirement Analysis | Feb-3-2016 | Started | Task 1 & preparing for the presentation. | Feb-22-2015 |  |
| Juan | Design | Feb-5-2016 | Started | Task 2 & preparing for the presentation. | Feb-22-2015 |  |
| Longhao | Design | Feb-5-2016 | Started | Task 1 & Task3 | Feb-22-2015 |  |

**Gantt Chart:**



**Gantt chart link:**

[https://app.smartsheet.com/b/publish?EQBCT=e7c83e7e7e4d4f96b9fa9bdac89140bf](https://app.smartsheet.com/b/publish?EQBCT=e7c83e7e7e4d4f96b9fa9bdac89140bf" \t "_blank)

## 6. Deliverables for each iteration

6.1. Iteration 1: Expected delivery date – 24 Feb 2016

1. We will be able to generate all the use case, process, interaction, class diagrams for our project.
2. We will build 30% of our application. This involves web page having functionality to add the events, view them and write notes.

6.2. Iteration 2: Delivery date – 23 March 2016

1. Modified use case, interaction, class diagrams as per the requirements.
2. We will be building 75 % of our application by Iteration 2. The features like saving notes entered by the user will be available. Adding pictures and also able to view them will be completed.

6.3. Iteration 3: Delivery date – 20April 2016.

1. We will develop all the application. Features like adding text to the picture will be available. Saving the events and the birthdays will also be available.
2. The application will be tested and available for demo.

## 7. Team qualifications

**Aparna Pavithran:**

I am a Spring 2016 Masters computer Science student at UT Dallas. I have completed programming in C/C++, Software Engineering, Object oriented Design and Analysis in UML, JavaScript, UNIX, Computer Network, Operating System, Data Structures, Algorithms as part of my previous academic studies. I have worked in two projects in JAVA and MySQL database server during my bachelor of studies. As part of industrial experience, worked on Healthcare and Telecom domain in Data Management field.

**Sriharsha Reddy Munjuluru:**

I enrolled in University of Texas Dallas in the Fall 2015 to pursue masters. My under graduation is in Computer Science and have 5 years of experience in IT field with various roles like developer, Consultant. I have good knowledge in programming languages like C, C++, Java, C# , scripting languages like JavaScript and JQuery , databases like Oracle and SQL Server. I undertook up to 6 projects during my experience and received rewards like Technical Expertise award and Project Star award.

**Amy Kiernan:**

I am enrolled as a Fast Track student in the Computer Science Master program, while also seeking to receive my Bachelor’s degree in Computer Science in May 2016. Through my education I have developed my programming skills primarily in Java, and through my internship I have learned C# and the .NET framework. I have experience in database design, developing APIs, as well as front-end work such as web development using AngularJS. A highly relevant class that I took was Software Engineering, where I worked with a group to extensively design and analyze the use of a language learning application.

**Sangwoo Moon:**

I had Computer science under graduation in Korea Military Academy in the year 2012 and currently pursuing Masters in University of Texas at Dallas. I have good skills in Java and C++. This is my profile:

* Undergraduate (Major, year): Korea Military Academy (Computer Science, 2012)
* Job experience: Republic of Korea Army officer (2012 ~ now)
  + Forster Student in UTD (2015~)
  + Vice Company Commander, 1nd Special Force Brigade Airborne (2014~2015)
  + Operation Officer, 7nd Infantry Division (2013~2014)
  + Leader of General Outpost, 7nd Infantry Division (2012~2013)
  + Platoon Leader, 7nd Infantry Division (2012)

**Juan Liu:**

I am enrolled in the Computer Science Master program of UT Dallas in fall 2014, so this is my last semester. As Java is my primary programming language, I have some knowledge on Object Oriented design and development. I also have experiences in database design and website development. From previous projects, I have learned how to take responsibility and collaborate with my teammates to achieve a higher goal.

**Longhao** **Gao:**

I am enrolled in the Computer Science master program as of Fall 2015 at UT Dallas. I have completed Data Structure and Algorithms and Operating Systems last semester. All projects that I worked on in these courses were developed using Java, so I have gained a good skill set using Java. Before 2015, I have done some 3D virtual real programs using C++, where I developed good practice in Object Oriented Analysis and Design. As part of industrial experience, I worked on Telecom domain in wireless communication protocol field.

## 8. Appendix - resumes

SriHarshaReddy Munjuluru Sangwoo Moon

 ****

Aparna Pavithran Juan Liu

 ****

Amy Kiernan Longhao Gao

**** 