Digitization of Government Systems:

Philippine Navy Seals Special Operations Group

Spring 2017 Final Report

Vertically Integrated Projects

Big Data for Public & Private Enterprises

Table of Contents

Client Background

The Philippine Navy
Point of Contact (POC): Lt. Casiple

The Philippine Navy (Hukbong Dagat ng Pilipinas or Armada Filipina) is the Naval warfare branch of the Armed Forces of the Philippines. It has an estimated 24,000 personnel and over 100 ships in active service. The Philippine Navy has the following powers and responsibilities:

- To organize, train, equip, maintain and operate naval forces and naval aircraft including naval reserve units, necessary to provide waterborne support and assistance required by the Armed Forces of the Philippines in the accomplishment of its mission
- To assist the proper governmental agencies in the enforcement of laws and regulations pertaining to navigation, immigration, customs revenue, opium, quarantine, fishing and neutrality in the territorial and contiguous waters of the Philippine Archipelago
- To develop, in coordination with the other major services and area commands the doctrines, procedures, and naval equipment for joint operations, and the doctrines and procedures for amphibious operations

Philippines Navy Seals Special Operations Group

With roughly 450 members, NavSOG is a battalion size group within the Philippine Navy. They operate nationwide, with their headquarters in the Cavite Naval Base (Cavite City). The group can be divided into two parts; base units and naval special operations units. Base units are involved with planning and logistics in the Cavite Naval Base, while the latter group are deployed on mission around the archipelago.

Soldiers in the unit can choose from four type units to specialize in; seal unit, special boat unit, diving, and EOD. Their naval special operations can be broken down into eleven units that focus on different regions of the country. Each operating unit has roughly 34 personnel from all four type units.

The Navy Seals Special Operations Group and their operations are used as a standard for local military excellence. The group is constantly looking to innovate and has recently been looking into integrating technology into their operations. The leaders of NavSOG have asked our team to conduct research into the current bottlenecks of their operations and provide them with tech-related solutions to these problems. Our goal is to develop scalable solutions that will eventually be held as gold standards by the broader Armed Forces of the Philippines.

Stakeholder Analysis

We have identified both the direct and indirect stakeholders affected by our project. Figure 1 shows that the soldiers and leadership of the Navy Seals special operations will feel the most immediate impact of our project. As we develop more complex features, this impact will extend to the Philippine Navy, the Armed Forces, and other government programs. The main reason for the potential extended impact can be justified by the role NAVSOG plays within the Philippine military and governance systems. As one of its most elite units, the AFP looks to NAVSOG as an operational excellence and innovation hub. Successful projects within the group have often been adopted by the wider Armed Forces and beyond.

Team Member Background

Member Contact Information

Name	Email
Stephen Go	sgo8@gatech.edu
Tu Le	ttc3@gatech.edu
Wai Man Si (Raymond)	wsi2@gatech.edu
Hamilton Wexler	hwexler3@gatech.edu
Darren Leung	darren.leung@gatech.edu

Original Problem

There are two main forms of merit within the Navy Seals Special Operations Group; getting promoted to a higher rank and being awarded scholarship opportunities. Although the group has a quantitative model for rank promotions, they do not have access to decision making tools for

scholarship awards. Their current system relies on the discretion of the management committee and is limited by the memories of officers present during the meeting. All soldier information is stored in physical files; leaders and administrators do not have any digital database to perform queries on. Since military education is an important component for the rank promotion model, the management committee wants to make more informed decisions by focusing on updated and accessible soldier data.

With **internal** and **external** threats on the rise, the Navy Seals Special Operations Group is seeking to set a <u>new standard</u> <u>for personnel and operations</u> <u>management</u> in order to maintain national stability.

| 2013 | 2014 | 2015 | 2016 |
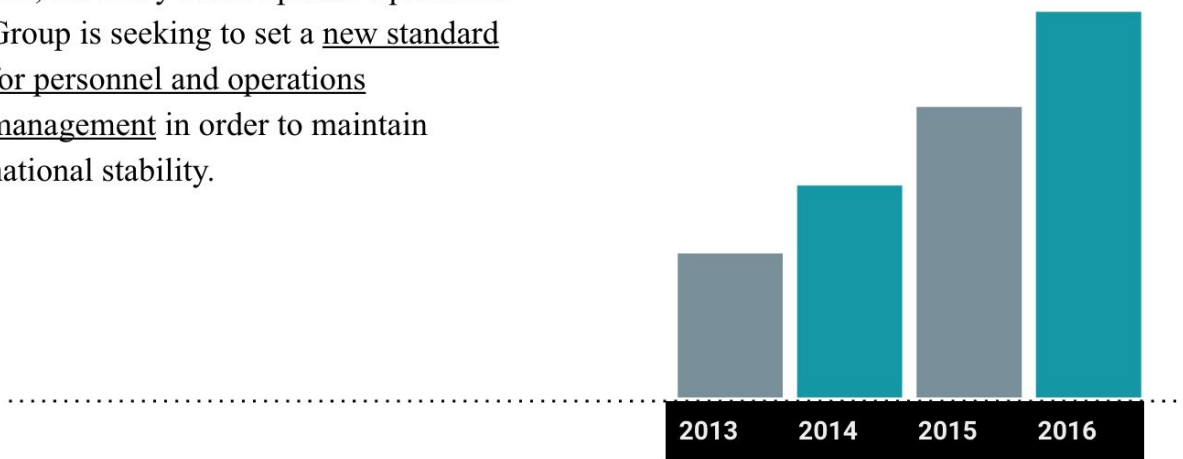
Project Background

Our goal is to develop a personnel information dashboard to aid the management committee in making decisions on scholarship awardees. Their current process involves the physical location of personnel files and sorting qualified records into a pile, which is then analyzed by leadership during committee meetings.

Assumptions and Constraints

- Information security is a main priority

- Input/output application

- There are key attributes or combinations of attributes that are favorable for scholarship decisions

Requirements Analysis

User Stories

We are using the Agile development framework for this project. One of the guidelines of this framework is a focus on feature analysis by user stories. User stories are feature preferences written by the different types of users of a product.

Below are some examples of user stories we collected from the group:

- As an operations analyst, I would like to have a central tool for storing and accessing soldier information.

- As a deputy commander, I would like to have a tool to filter soldiers by different experience parameters.

- As a soldier, I would like to know that my experiences are being considered.

These stories served as the basis of our initial feature decision-making process. We found that the integral features to the group were related to digitization of current systems, understanding their current workforce, and comparison tools for decision-making. Although we are developing the initial application for their administrative personnel, we found it useful to collect stories from soldiers holding a broad range of positions to get a holistic perspective on their current and future needs. These stories were also used in the development of our feature roadmap below.

Attributes

Through an initial interview, some personnel attributes include the following:

Attribute	Sample Values	Notes
Name	John Doe	-
Age	(integer)	How is age taken into account?
Years in Service	(integer)	-
Civilian Education	HS, Undergrad Local, Undergrad Int'l, Grad Local, Grad Int'l, Post-grad Local, Post-grad Int'l (categorical)	Is the school name important for decision-making? If not, do you consider level of education?
Military Education	Commissioning course, basic, advanced, GSC - general staff college, MNSA (categorical)	Do certain programs have prerequisites? Is there a ranking of military education programs?
Officer/Enlisted	Officer or Enlisted (binary)	-
NavSOG Class	year	-
Positions Held	Team Leader, Ex-officer, Commanding Officer (list)	What are all the possible positions? Do you have a ranking for them?
Type Unit	Diving, Seal, Special Boat Unit, EOD (categorical)	Does the Type Unit affect scholarship decisions?

Development Checklist

1. Analysis of attributes - understand the relative priorities of soldier attributes and how they relate to scholarship.

2. Initial System Design - develop wireframes of the system.

3. Dashboard Components - justification for dashboard design and components.

4. Sprint Backlogs - detailed schedule of deliverables.

5. Conduct usability studies

6. Iterate and refine product

7. Benchmark project impact

Nielsen Norman Usability Heuristics

Visibility of system status

The system should always keep users informed about what is going on, through appropriate feedback within reasonable time. We plan to incorporate this in our application by including visual pop-up feedback for successful or failed operations.

Match between system and the real world

The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order. In order to abide by this principle, we intend to conduct process mapping of the various operations that the group conducts.

User control and freedom

Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo. We plan to provide administrators with high degrees of freedom by allowing them to filter personnel by different attributes.

Consistency and standards

Users should not have to wonder whether different words, situations, or actions mean the same thing. Although our application's initial design will remain simple, we plan to take note of any convention used in order to maintain consistency as development progresses.

Error prevention

Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action. For our application, we have incorporated client-side validation, which highlights errors before they are made.

Recognition rather than recall

Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate. We have encouraged recognition over recall at every point of development. One instance of this can be seen on the "view personnel" page, where we used a pencil icon to denote the functionality of editing soldier information. Since we have assumed a low level of technical literacy, this principle is one of the most important to our development decision-making.

Flexibility and efficiency of use

Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions. Since our target users will not be experts, we have not included any form of accelerators into the design of Phoenix 1.0. As we observe improvements in their expertise with respect to our system, we may include various shortcuts to improve their user experience and further increase efficiency.

Aesthetic and minimalist design

Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility. For our application, we made sure not to include any irrelevant information; every page contains the minimum amount of text and functionality to facilitate ease of use.

Help users recognize, diagnose, and recover from errors

Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution. We plan to incorporate error messages for any failed task, making sure that the end users know the mistakes they have made and are able to correct them. In addition to modals, we have incorporated client-side validation; this alerts users of mistakes before submission of tasks. As a result, they are able to recognize mistakes at a much faster rate compared to other alternatives.

Usability Study

Methodology

We plan to use Heuristics evaluation, a form of usability engineering, for finding the usability problems in a user interface design so that they can be attended to as part of an iterative design process. Heuristic evaluation involves having a small set of evaluators examine the interface and judge its compliance with recognized usability principles

The evaluation will be conducted by multiple evaluators as different people would find different usability problems. Heuristic evaluation is performed by having each individual evaluator inspect the interface alone. Only after all evaluations have been completed are the evaluators allowed to communicate and have their findings aggregated. This procedure is important in order to ensure independent and unbiased evaluations from each evaluator. The results of the evaluation can be recorded either as written reports from each evaluator or by having the evaluators verbalize their comments to an observer as they go through the interface.

Different from traditional user testing in which the experimenters are reluctant to provide help when the users make mistakes, heuristic evaluation sessions, heuristic evaluation test a domain-specific application, experimenters are more willing to answer the evaluators' question about the domain because evaluators might not be domain experts. Moreover, the process of answering the evaluators' questions will enable evaluators to better assess the usability of the user interface with respect to the characteristics of the domain. However, the evaluators should not be given help until they are clearly in trouble; otherwise, the testing results might be biased.

During the evaluation session, the evaluators go through the interface several times and inspect the various dialogue elements and compare them with a list of recognized usability principles.

Procedure

Before

For the most beneficial and informative results, the environment of the usability testing session should be casual and natural. Do not use terms such as 'usability testing' or 'market research' when inviting experimenters to conduct the testing, as these can confuse and put people on edge. Also, ensure that participants know how long the usability testing will take, and the type of tasks they will be expected to perform.

During

The experimenters were requested to perform a list of assignments or tasks that related to the functionality of the tested application. Experimenters observe the process and record the experimenters' performance. The record includes the time it takes for the experimenters to perform a specific task, their vocal and facial responses during the interaction with the application. Notes of any terms or phrases they use were taken- this not only demonstrates you take their feedback seriously but may provide useful tips as to possible labels for key functionality or navigation.

Rather than simply ordering test users to do tasks without explanation, the experimenters situated the request within a short scenario that sets the stage for the action and provides a short explanation and context for the purpose of the testing.

Tasks and Surveys Used

Before writing the task scenarios used in the testing, we created a list of general user goals that were applicable to our application. It is useful to ask the question: what are the most important things that every user must be able to accomplish on the application.

Aside from assigning tasks to experimenters, the evaluators ask the user to suggest tasks. While this gives another indication of their expectations and requirements, it may suggest new functionality or priorities. The tasks mentioned below make up a comprehensive list of capabilities of the Phoenix 1.0 application before meeting with the NavSOG in the Philippines, as well as the surveys we used to gather further information from the primary end users.

Task 1:

 Goal: Add a new personnel

Task: There is an influx of new members after the NavSOG initiation week. As a system administrator, you are required to enter their information into the Phoenix application. Locate the function and add all new members into the system.

Task 2:

Goal: View information of a personnel

 Task: Now you are a new Navy Seal officer and you want to check your information on the database. View your profile in the application

Task 3:

Goal: Login

Task: You are now a Navy Seal administrative personnel and you want to use the application. Log in to the Phoenix application.

Task 4:

Goal: Query a personnel

Task: The lieutenant is interested in finding 10 officers for are qualified for the annual scholarship for an international education program. The basic requirement for the scholarship is the officer has served Navy Seal for more than 5 years. Search for a list of officers that qualify this condition.

Task 5:

Goal: Evaluate a team using the dashboard

Task: The lieutenant is interested in assessing current team health and effectiveness.

Evaluation Findings

Initially, the Phoenix application was a desktop application with several different views of the data. A desktop application was initially decided on for increased data safety and to better accommodate the varying levels of technical knowledge of end users. However, after the Heuristic evaluation was complete, Phoenix was changed to a web application to give the end users a more familiar experience. End users felt more comfortable with a web application style based on their previous experiences and as we found that the Navy Seals Special Operations

Group preferred pure data visualization over the ability to switch between the raw data and data visualization functions. Additionally, the shift to a web application allows the data to be viewed by more personnel and increases availability of the data as the officers who manage the personnel may have to move from place to place frequently. As the evaluations continued, it became apparent that the focus of the application needed to be shifted more from simply management for education and training decisions to more mission critical decisions (such as when to rotate out certain soldiers due to being in the field too long or making sure teams in the field are properly balanced across team roles), the mobility of the application also increased in importance because the time sensitivity of decision making increased.

Revised Problem

After meeting with the Navy Seals Special Operations Group, as mentioned earlier, the objective of our project was changed from decision making in relation to education and training to more mission critical decisions. Mission critical decisions include when to rotate out personnel to provide rest for soldiers, developing balanced teams, and more. By decreasing the time it takes to make these decisions and giving data based insights for improved decisions, our data product will help improve military operation efficiency and effectiveness.
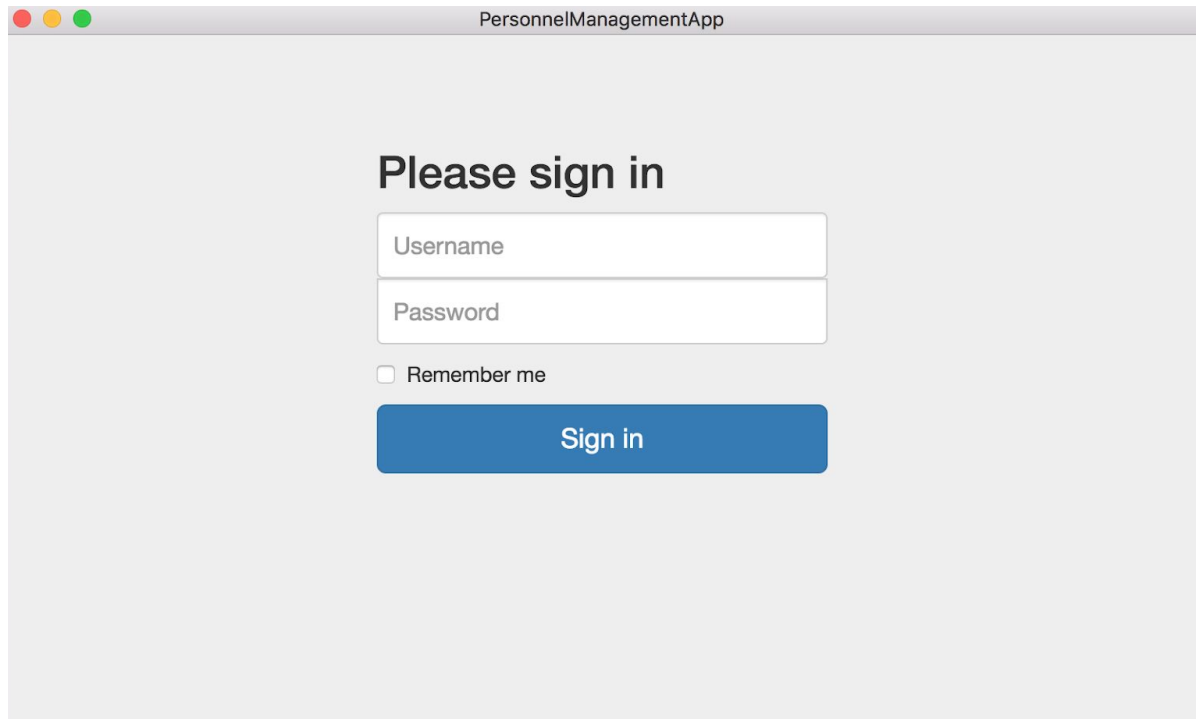
Current Product

Product Features

To solve the current problems of the Navy Seals Special Operations Groups, we have developed a basic personnel management system. The product is a cross-platform desktop application that provides the group with three main capabilities; to digitize current records,

analyze trends and personnel structure, and comparison of personnel for promotion decision making.

Login Page

Figure 3: *Phoenix 1.0 _ Login Page*

The login function is used to identify and authenticate users. The initial application will be used by administrative personnel, but we will maintain the ability to assign different access levels to users to protect sensitive information at scale. In addition, it can create an event log for each user to keep tracking their access history for security purpose. The event log file will help us to diagnose problems in the system.

Main Page
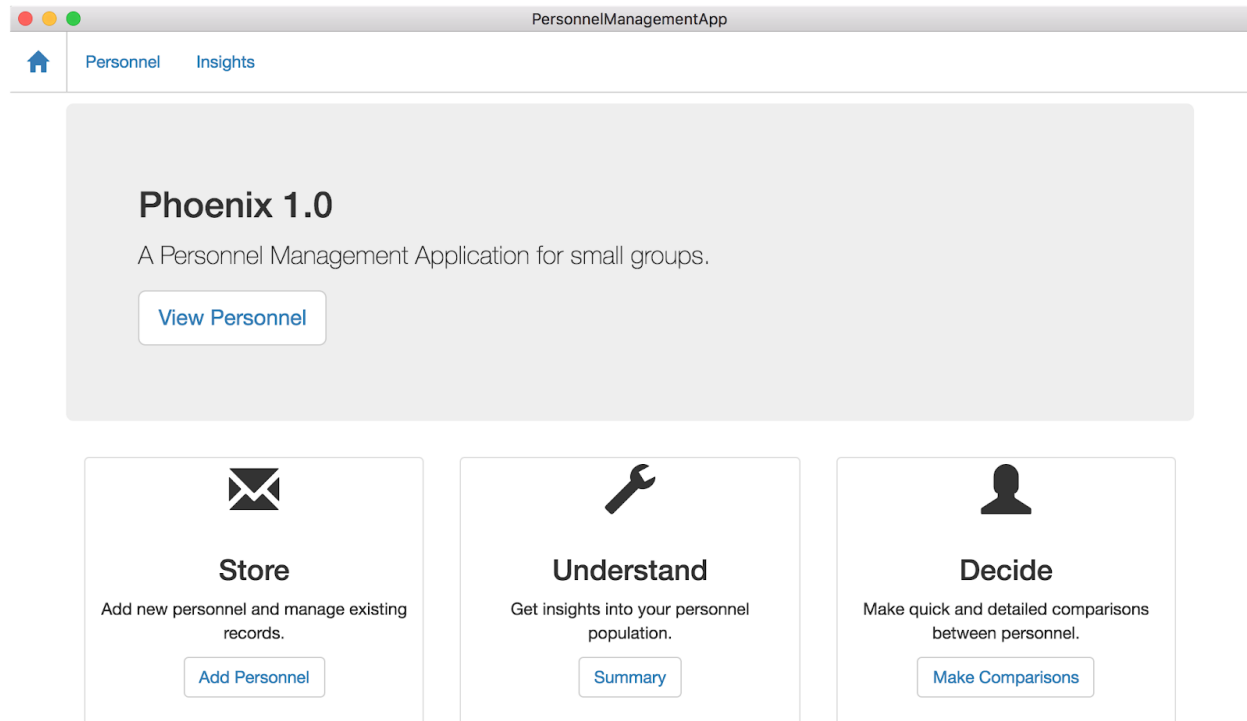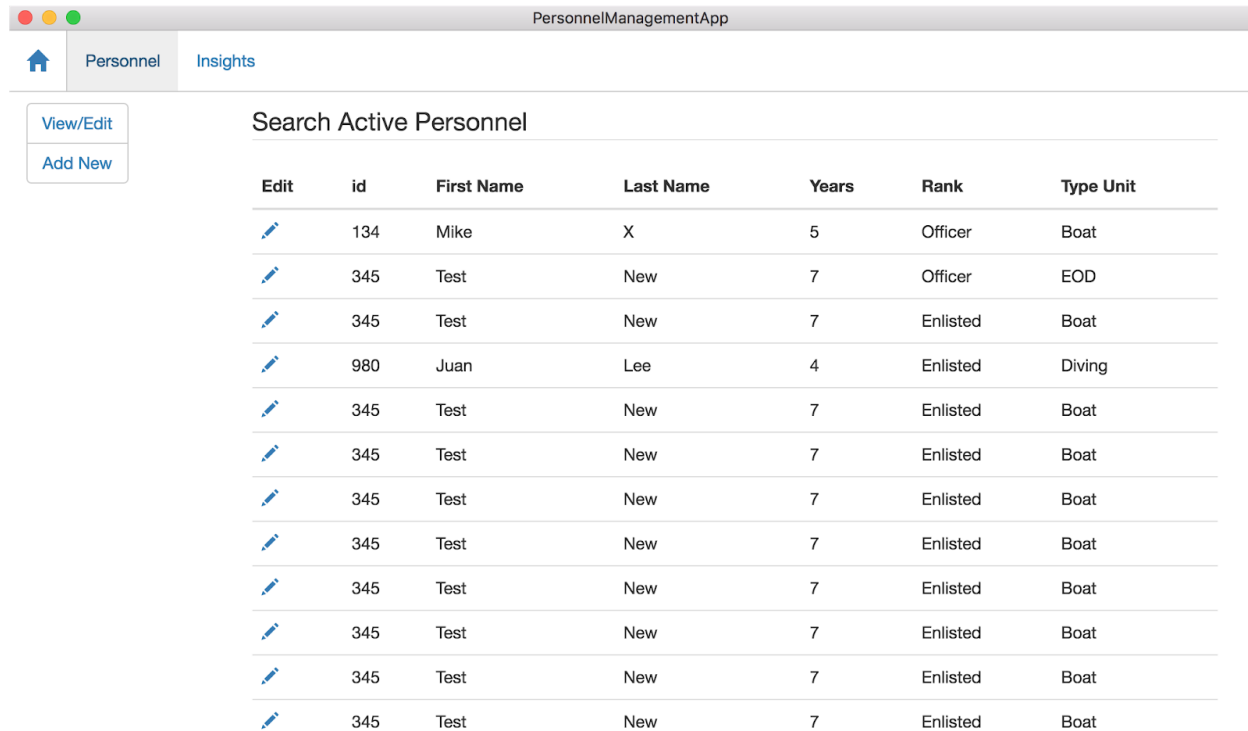
Figure 4: *Phoenix 1.0 _ Main Page*

In order to promote usability and clearly identify the affordances of the Phoenix system, we have developed the main page to directly showcase the capabilities of our application. Links to the four main features are outlined; view personnel, store information, understand their current human resource infrastructure, and make decisions through personnel comparisons.

Through various interviews, we qualitatively determine the technical skill level and access of the leadership and operating personnel of the Navy Seals Special Operations Group. We have been developing the product under the assumption that the various personnel in the group do not work with technology on a daily basis. Therefore, it was imperative that we designed the application to cater to novice users.

View Personnel Page

Figure 5: *Phoenix 1.0 _ View Personnel Page*

For the view personnel page, we have developed basic functionality to fill tables from local databases. This has two unique and critical benefits for the group; minimized security concerns and easily accessible information. We have also included an image of a pencil to allow administrative personnel to edit information from the page itself. This promotes the usability principle of recognition over recall.

One important feature that we plan to add to this page is a filter that will allow NavSOG's leadership to divide their workforce into useful subsets that will give them a better understanding of their current workforce. In order to integrate this feature effectively, we have planned to do more research into the various needs of the local leadership. Some useful information we need to move forward include:

- Relative importance of each tracked feature

- Other useful attributes outside of our current list

- Their current evaluation heuristics

- Post-decision evaluation

From our initial evaluation, we found this area to be the most relevant to reducing their current decision-making process. The main complaint we received from the group involved filtering of personnel by qualification.

Add Personnel Page

Figure 6: *Phoenix 1.0 _ Add Personnel Page*

The "add personnel" interface of our application will be used as the main point of the digitization of current soldier records.

Dashboard

For the Dashboard, we focus on the management on the whole squad instead of individual.
There is five features for showing you the detail of the squad: Cities, Average Time To Leave,
Type Unit, Experience and ratio of officer and enlisted. By clicking on those feature, you can
filter those information that you don't need.

During our visit to the Navy Seals HQ, we learned that their team was interested in getting a
general overview of their teams. Due to the decentralized and varied nature of missions around
the country, the management committee

The dashboard will have two main filters, located on the top left (map) and mid left (type unit)
sections. Since the main purpose of the interface is to provide information on teams, we used a
map of the Philippines to display all deployed teams around the country.

Clicking on the city of a team will filter the information to show the commanded relevant indicators for the selected team. The filters can be nested, added, and removed based on the immediate specifications of the management committee.

Development Stack

1. Github

   Github allows us to track the changes we make to our code and it is good for version control.

2. Electron

   Electron is a framework for creating cross-platform native applications with web technologies like JavaScript, HTML, and CSS. Electron provides a runtime to build desktop application by taking a main file defined in available package file and executing it. The file would create application windows which contain rendered web pages with the added power of interacting with the native GUI (graphical user interface).

2. Ember-CLI

   Ember-CLI is a web application framework; combined with the electron framework, it allows us to run our program as a web application and a desktop app. Furthermore, it provides standard testing integration.

3. D3.js (3.x version)

   The D3 framework allows us to develop heavily customized visualizations and dashboards for our client's needs. For example, a radar chart can be used to represent the soldier's information.

4. ADDEPAR

   It is a open source project which allows you to render very large data sets by only rendering the rows that are being displayed. Also, it includes a charting library built with the Ember.js and d3.js frameworks.

5. Ember Google Charts

It allows you easy to implement Google Charts in Ember CLI apps.

6. Tableau

For our dashboard prototyping, we use the Tableau software.

Feature Timeline

Although the initial product focuses on personnel management, we have identified several operational areas that will be useful to the group in the foreseeable future. A natural extension to the current application is related to measuring recruitment strategies and providing the group with insights into how they can efficiently expand their footprint in the country.

Furthermore, we believe the application can be extended to include various operational information on the group. We plan to integrate various external data sources to help the group improve their decision-making process for broader facets of strategy, such as mission priority development. We also plan to include simulation and optimization capabilities into the application to provide the group's leadership with both predictive and prescriptive affordances that will allow them to maximize the use of unstructured data sources.

Data-centric product development plan over the next 5 years.

Figure 8: *Data-centric Product Development Plan over the Next 5 Years*

Conclusion

We have presented our application, including the dashboard for team management, to our sponsors. They are currently testing the dashboard and learning about the potential impact of data science on their operations. In addition, we have provided them with user documentation, development documentation, and a plan of action for future development of the application.

The unit leaders were impressed with the current dashboard, citing its ease of use and powerful insight as bonuses. During the interface test, all subjects completed tasks (above) in less than a minute and did not require any training beforehand.

References

Nielsen, Jakob. "10 Usability Heuristics for User Interface Design" . January 1, 1995. *Nielsen*

Norman

      Group. https://www.nngroup.com/articles/ten-usability-heuristics/. Accessed December

7, 2016

Nielsen, Jakob. "How to Conduct a Heuristic Evaluation". January 1, 1995. *Nielsen Norman*

Group.

      https://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/. Accessed

December

      7, 2016

Dunn, Tory. "8 Guidelines for Usability Testing". April 1, 2016.

      http://www.webcredible.com/blog/8-guidelines-usability-testing/. Accessed December 7,

2016