Project Deliverable C: Conceptual Design, Project Plan, and Feasibility Study

GNG 2101 – Intro. to Product Dev. and Mgmt. for Engineers

Faculty of Engineering – University of Ottawa

Submitted by

[B34]

[Michael Kagnew, 30011347]

[Fatimah Vakily, 300125671]

[Gabriel Cordovado, 300110852]

[Joseph Francis, 300116730]

[Jasmine Kokkat, 300115249]

[Emma Ballantyne, 300115563]

October 1, 2020

University of Ottawa

Table of Contents

Introduction	3
C.1 Conceptual Design	3
C.2 Project Plan	14
Tasks:	14
C.3 Feasibility Study	15
Conclusion	16

Introduction

The purpose of this deliverable is to utilize the previous deliverables and client meet information in determining what ideas should be generated, and ultimately used, that will satisfy the requirements given in deliverable A and B.

The beginning portion of the deliverable focuses on conceptual design ideas from all group members in order to tackle different parts of the application. By splitting the application into functional components, ideas were easier to generate. The team decided to have at least 3 concepts per member, which would then be trimmed down to much fewer ideas to continue exploring.

This deliverable also required updates to our project management system (i.e Gantt table), which is needed to figure out what tasks are at what stage, and by whom.

The deliverable is completed with a feasibility study using the TELOS factors. The team discusses any concerns and potential risks that are foreseeable. Each section of the TELOS factors is discussed and outlines how we will move forward to ensure our project will be successful.

C.1 Conceptual Design.

Based on customer needs, the required functions include recognition of a notification, user interface, reading notifications, storing incoming and missed notifications, and user privacy.

To recognize what a notification is, the application needs to differentiate images from text and send it back to the user through voice commands.

The user interface can have a minimalist widget-approach and have the simple click-navigation rule. The application would ideally use SiriKit to manipulate app-settings with Siri voice commands. The interface can use bright colours over pastels so that text is easier to recognize. The application can avoid widgets with "compressed" elements such as Rolodex's, which are scroll selectors. This application is ideally able to be customized for the client to enable and disable application and adjust sound settings. The user can then be able to do other tasks without having disrupted functionality from the application.

The application can read notifications with the ability of having multiple language options. It can read who the notification is from, which application the message is from, and the entire message. A stop function will be implemented to prevent additional message information being read at every use.

Another priority when making our application is to store incoming and missed notifications. The data stored on the phone include the app name, time of the notification (current time – time on screen), the sender's name, message blurb, and number of pending messages (if stacked on-top of each other). The application can store notifications in order and only read aloud once user asks to access missed notifications.

For user privacy, we plan to wipe stack/queues/similar structures for storing missed notifications after given N time specified within the settings. After extracting notification data from a screenshot/capture, we can delete the photo, as useful data would have been stored in the queue. We intend to enable "app states" that change the data elements spoken out loud such as home, away and busy. Home would include details like app name, time, individual, and the message blurb. The away setting could include app name, time, and individual. When the user is busy, nothing would be notified.

2. Group Concepts:

Fatimah: Aesthetics [like app button visual design]

- 1) minimalist design; simple fonts, intuitive layout and clear contrasts
- 2) allow users to change text size within the application
- 3) night/day mode, invert colours options for the app

Jasmine: OS interactions

1) Loads the application based on the voice commands sent to the operating system.

2) Uses APIs to access the resources the application needs to start running and return the correct information.

3) Once the user exists the application, the OS will temporarily suspend the application so that memory is cleared to prepare to run another application.

Emma: User Interface

- 1) Allows the user to enable and disable the application
- 2) Ability to mute the voice function and adjust the sound settings
- 3) Implement the 3-click navigation rule

Joseph: Security

- 1) User end security is the focus as our app should only communicate between apps and the OS
- 2) User can force stop the application reading messages if there is sensitive information and the user is in a public place
- 3) Automatically delete stored notifications after x days (as set by the user)
- 4) Settings cannot be accessed via voice when the phone is locked. By default, saved notifications cannot be accessed when the phone is locked but this setting can be changed

Gabe: Voice commands

- 1) Control UI-components and application settings (language, etc.) through Voice-capture.
- 2) Implement SIRI shortcuts for interaction with Backend, outside of the application.
- 3) Trigger text-to-speech on stored notifications captured on the device locally or in storage.

Michael: Database/Storage

- 1) Store screenshots into a queue system and have the API analyze in real-time
- 2) Pass along screenshots into Amazon's DB system (allow them to analyze and store)

3) Store the text that is analyzed from API in SQL database system to allow for future backtracking

3. Analysis of Group Concept Choices:

Aesthetics:

	Target specifications			
Concept	Intuitive button	Customer	Ease of	Ease of gesture
	usage to	Satisfaction	installation and	navigation
	activate/deactivate		setup on phone	
Minimalist	4	4	4	4
design; simple				
fonts, intuitive				
layout and clear				
contrasts				
Allow users to	1	4	4	4
change text size				
within the				
application				
Night/day	1	3	4	4
mode, invert				
colours options				
for the app				

Reasoning/logic:

The minimalist design was something specifically mentioned by our client as a feature wanted for the application. The suggestions allow for ease of installation, setup, and gesture navigation as they grant the user the ability to modify the features of the app's appearance to whatever suits them best without complicating the function of the app.

OS interactions:

	Target specifications			
Concept	Voice controls display information from recent notification	Non-disruptive functionality to phone usage	Multi-language use (English and Spanish)	Reads only notification- related information
Loads the application based on the voice commands sent to the	4	4	4	4

operating				
system.				
Uses APIs to	4	4	3	4
access the				
resources the				
application				
needs to start				
running and				
return the				
correct				
information.				
Once the user	3	2	2	3
exits the				
application, the				
OS will				
temporarily				
suspend the				
application so				
that memory is				
cleared to				
prepare to run				
another				
application.				

All of the concepts offer solutions that meet the target specifications listed, as they follow the use of voice commands, access and return only relevant information, and can be adapted to support multi-language use without disrupting the regular functionality of the user's phone.

Additional Features:

	Target specifications		
Concept	Non disruptive functionality to phone usage	Ease of gesture navigation	
Allows the user	3	4	
to enable and			
disable the			
application			
Ability to mute	4	3	
the voice			
function and			
adjust the sound			
settings			
Implement the	3	4	
3-click			
navigation rule			

Reasoning/logic:

These two specifications are related to the concepts because each of them defines the first thing the client interacts with. The values are decided on a mixture of empathy thinking (i.e. determining how the client would feel about these concepts) and determining how the concept stacks up against the specifications.

Security

	Target specifications
Concept	Reads only notification-related information
User end	4
security is the	
focus as our app	
should only	
communicate	
between apps	
and the OS	
User can force	4
stop the	
application	
reading	
messages if	
there is sensitive	

information and the user is in a public place	
Automatically delete stored notifications after x days (as set by the user)	4
Settings cannot be accessed via voice when the phone is locked. By default, saved notifications cannot be accessed when the phone is locked but this setting can be changed	2

One requirement that directly relates to the security of this application is "reads only notification-related information" because the concepts are designed to be hidden from the user. The values were determined by deciding what concept would best reinforce user security.

Voice Commands

	Target specifications			
Concept	Level of User Control	Multilingual support	Customer Satisfaction	
Control UI-	4	2	3	
components and				
application				
settings				
(language, etc.)				
through Voice-				
capture.				

Implement SIRI	3	4	4
shortcuts for			
interaction with			
Backend,			
outside of the			
application.			
Trigger text-to-	2	3	4
speech on			
stored			
notifications			
captured on the			
device locally or			
in storage.			

Voice-command concepts were derived primarily from the client meeting. Hence, the values and target specifications are based on the values of the client and empathy of the team towards the situation. Negative connotations to points are applied based-upon technological limitations and historic rates of error.

Database and storage

	Target specifications
Concept	Time to load application
Store	4
screenshots into	
a queue system	
and have the API	
analyze in real-	
time	
Pass along	3
screenshots into	
Amazon's dB	
system (allow	
them to analyze	
and store)	
Store the text	3.5
that is analyzed	
from API in SQL	
database system	
to allow for	

future	
backtracking	

No direct way to viably determine target specifications, since database system is unknown to end user; however, performance of application can be felt. Storing data in a queue system eliminates the need to hold the data long term, thus only having little data being stored, thus being the fastest. Sending data to Amazon is more time consuming and requires a reliable internet connection since all processing and storage is handled on the cloud, away from the local machine. Storing text in an SQL database allows for the user to view old notifications; however, this will slow down the application over time.

4. Promising Ideas to Build On

Aesthetics: Minimalist design; simple fonts, intuitive layout and clear contrasts

OS interactions: Loads the application based on the voice commands sent to the operating system.

Additional Features: Implement the 3-click navigation rule

Security: Automatically delete stored notifications after x days (as set by the user), User can force stop the application reading messages if there is sensitive information and the user is in a public place

Voice Commands: Trigger text-to-speech on stored notifications captured on the device locally or in storage, Implement SIRI shortcuts for interaction with Backend, outside of the application

Database and storage: Store screenshots into a queue system and have the API analyze in realtime, Store the text that is analyzed from API in SQL database system to allow for future backtracking

5. Group design concept:

An application with a minimalist layout that functions based on receiving voice commands from the user and operates using the 3-click navigation rule, though users can set certain security

functions to ensure their privacy. The voice commands use text-to-speech on sorted notifications, and screenshots are stored in a queue system. This concept seems most desirable at this point in the design process based upon the knowledge we gained from discussing with our client and the target specifications of deliverable B, and most realistic based on the limitations given by Apple for outsiders designing iOS apps.

,		Radi	Applicasion-	
	Stored Messages	2 Languge Capability	Widgets	6 10
Privacy Protocols	Re	ad ticications +	UI.	Cocevity 1
	R		Hute Function	
	Ap	pPiotocol3	Voice Regignition]
				Hilbory

6. Visual Representation



C.2 Project Plan

Tasks:

this is a short-term task list denoting the immediate problems facing us as the project stands *now*. (For the updated long-term strategy see the lower image and notes):

- 1. Develop Framework for app and basic interactions with OS
 - a. Build framework actualizing Gabe's framework
 - b. Research protocols to connect app to Siri and voice interaction
 - c. Research iOS security protocols to ensure there are no unexpected issues
 - d. Plan the features for a 'skeleton app'
- 2. Meet with Client
 - a. Present possible features and their rankings
 - b. Receive client feedback
 - c. Use the feedback to update the project plan
- 3. Preliminary Drafts
 - a. Draft the code for an "skeleton app"; this app has some of the most basic functionalities
 - b. Test the skeleton app and connect it to Siri
 - c. Review the app and note any problems and possible improvements
- 4. Detailed Design
 - a. Use the preliminary drafts and construct a full plan
 - b. Begin drafting the new design

Gnatt Chart of Task w/ Responsibilities:

Menu 🗸 View 🗸		Everyone 🗸 🛛 All						Dates	s 🗸 All Colors					✓ □ Hide Completed								
	Assigned	Progress																		осто	OBER	202
Client Meeting (All)			24	25 2	26 2	27 28	3 29	30	1	2	3	4 5	6	7	8	9 10	0 11	12	13	14 15	5 16	17
Receive Client Feedback (Joseph)		0%	т	F	S S	S M	Т	W	т	F	S	S M	Т	w	т	FS	S	М	Т	W T	F	S
Update the new Project Plan (Joseph)		0%									Ċ		h									
<u>Task</u> <u>Milestone</u> <u>Group of Tasks</u>																						
 Detailed Design 		0%																				
Construct a Detailed Plan with all major features (all)		0%												-	-0							
Establish BOM (Joseph)		0%																		Ċ		
 Develop Framework 		0%											Þ									
Research iOS Security Protocols (Joseph		0%																				
Research Voice Protocols (Jasmine)		0%																				
Research Protocols to actualize Gabe's Framework (Gabe)		0%												4								
Task Milestone Group of Tasks																						
 Preliminary Drafts 		0%																				
Review App and any problems (Michael)		0%																				
Test Skeleton App with Voice (Jasmine)		0%																				
Draft A skeleton App (Fatimah & Michael)		0%										Ģ		1								
Plan a Skeleton App (Fatimah & Michael)		0%																				
C Task Milestone Group of Tasks																						

Long-Term Plan

The long-term plan as shown below has been updated, with old tasks removed to allow us to focus on the present, and times updated to reflect changes in the project. The current critical path is quite clear,

with 3 parallel summary tasks being completed separately to lead towards the milestone deliverable D. Dependencies have been optimized to allow us to complete these tasks independently until they have to converge at a given date. Later tasks will be fleshed out as Deliverable D is completed, since as of now we do not have a clear idea of all development goals. (Please ignore the weird tasks at the bottom, showing recurring tasks can give some issues in MS Project).

	<u>ن</u> - رک	c) - +		Gantt Charl	t Tools			GNG2101_D1_PartC - Project Professional Joseph	rancis 📧 — 🗗 🗙
File	Tas	k Resi	ource Report Project View Help	Forma					
Gantt Chart View	Pas	te Clipbe	rt Calibri • 11 • pry * srmat Painter oard Font 5	× 25× 50× 75× - → → 0	mark on Respect i 20 ↔ ↔ Inactivat Schedule	Track - Links Manu e Schee	ally Auto Jule Schedule	bageet Nove Mode Tasks Summary Milestone Deliverable Tasks Inset	~
TIMELINE	Sat 2	Start 0-09-19	'20 Sep 20 '20 Sep 27 '20 Oct 0	4 .50 0	Dd 11 '20	Oct 18 '21	0 Oct 25	20 Neve01 [20 Neve01 [20 Nev 15 [20 Nev 22 [20 Nev 29 [20 Dec 66 [20 Dec 13 [20 Dec 20] Add tasks with dates to the timeline	21 Jan 03 Finish Sat 21-01-09
	0	Task Mode	▼ Task Name	- Duration	- Start -	Finish •	Predecesso	2020 CBr 3 2020 CBr 4 2021 CBr 1 202 esso Jul Aug Sep Oct Nov Dec Jan Feb Mar	Apr May
1		*	PD C	0 days	Thu 20-10-01	Thu 20-10-01		• 10-01	
2		*	Client meet 2	0 days	Thu 20-10-01	Thu 20-10-01	1	10-01	
		*	Develop Framework	8 days	Thu 20-10-01	Thu 20-10-08	2		
4		*	Research Security Protocols	6 days	Thu 20-10-01	Tue 20-10-06	2	Joseph	
5		*	Research Voice Protocols	6 days	Thu 20-10-01	Tue 20-10-06	2	Jasmine	
(*	Research Documentation to Actualiz Gabe's Framework	e 6 days	Thu 20-10-01	Tue 20-10-06	2	Gabe	
		*	Preliminary Drafts	8 days	Thu 20-10-01	Thu 20-10-08	2	ř –	
8		*	Draft a Skeleton App	6 days	Thu 20-10-01	Tue 20-10-06	2	Fatimah	
S		*	Test interaction w/ voice protocols	2 days	Wed 20-10-07	Thu 20-10-08	8	i Jasmine	
1		*	Review the Skeleton and note any problems	2 days	Wed 20-10-07	Thu 20-10-08	8	👔 Fatimah	
1		*		15 days	Thu 20-10-01	Thu 20-10-15			
b 1		*	Establish BOM (if any)	6 days	Thu 20-10-01	Tue 20-10-06		Emma	
1 CHAI		*	Construct a Detailed Plan with all major features	6 days	Thu 20-10-01	Tue 20-10-06		Michael	
L 1		*	PD D	0 days	Fri 20-10-09	Fri 20-10-09	10,9,6,5,4,	5,4, 😵 10-09	
9 1		*	PD E: Project progress presentation	0 days	Thu 20-10-15	Thu 20-10-15		↓10-15	
1		*	Client meet 3	0 days	Wed 20-10-21	Wed 20-10-21	15	* 10-21	
1		*	PD G	0 days	Thu 20-10-22	Thu 20-10-22	16	<u>↓</u> 10-22	
1		*	Prototype 2 (Research & Prelim)	7 days	Wed 20-10-28	Tue 20-11-03	17	Michael	
1		*	Prototype 2 (Drafting)	7 days	Wed 20-11-04	Tue 20-11-10	18	Joseph, Fatimah	
2		*	Prototype 2 (Testing)	7 days	Wed 20-11-11	Tue 20-11-17	19	Gabe	
2		*	Prototypez (Analysis& Benchmarking)	/ days	Wed 20-11-11	Tue 20-11-17	19	Joseph 13.02	
2		-	FUI Einel mototume	o days	111U 20-12-03	Thu 20-12-03	21.20	Gaba	
2		×	Intensive Drafting	15 days	wed 20-11-18	wed 20-12-02	21,20	Gaue Gaue Satisan	
2		5	Renchmarking and Troubleshooting	7 days	Wed 20-11-18	Wed 20-11-24	21,20	Jacmine Michael	
2		-	Design Day	1 day	Thu 20-12-03	Thu 20-12-02	23.25	Joseph Fatimah Gabe Jasmine Michael	
2		*	Monitoring and Control	1 001	110 20-12-03	110 20-12-00	20/20		
2	0	*	Project Objectives	78 days	Sun 20-10-04	Sat 21-01-09			
4	00	*	Quality of deliverables	57 days	Mon 20-09-28	Sat 21-01-02			
4							Þ		Þ
Ready	*	New Tasks	: Manually Scheduled						
-	9	Type her	re to search	0 🛱	ः 😸	a			(A \$4) ENG 2020-10-01 €

C.3 Feasibility Study

This project has inherent uncertainties, more so than risks, due to the nature of this project. The uncertainty comes from a few key areas, namely our lack of skill in application development, iOS limitations, and programming environment issues.

Technological

Overall, our group has limited experience when it comes to application development, and since our client uses iOS, our only real option for development is using Xamarin which uses the language C#. The software is powerful; however, no one in our group has experience with it, thus it can be cause for concern. In addition, iOS requires us to use an Apple computer in order to compile the project, and since only two people in our group have an Apple computer, we will need to find a way to rent one over the cloud that will support multiple remote in users.

Economic

Our project has a maximum budget of \$50 that has been set by our client. We are planning to use Xamarin to develop our application which is a free, open-source platform and therefore will aid in keeping our costs low. However, due to the fact that our client needs an IOS application we need to use apple computers to compile our code. Only two out of our group of six people have access to an apple computer, therefore we have been looking into renting an apple computer. The rental would cost \$118 for four months giving us access for 8 hours a day. This is our only anticipated project cost; however, it would put us \$68 over our proposed budget. We have a meeting scheduled with our Professor to discuss the possibility of altering our budget or being provided with additional resources.

Legal

At this point in our planning there are no anticipated legal barriers. The genre of application we are developing is legal and there are various similar applications already on the market. We have taken into account security and privacy issues by having including various functions to help uphold the required standards. These functions include a force quit function to avoid sensitive information being read in public as well as including a function to automatically delete the screenshots that are taken by the application after a pre-set amount of time.

Operational

To ensure the successful operation of our project our team needs to ensure that everyone reviews C# to be able to write our program in this language. We also require apple computer resources to ensure our code can be compiled, as discussed in the economic section. Finally, our operational success relies heavily on frequent, clear communication between all team members, as well as with the client. This is being upheld by our frequent group meetings and scheduled client meetings.

Scheduling

To remain organized throughout the project our team is using Microsoft Projects as our hub for organization and scheduling. We have a detailed short-term tasks list to plan out the immediate problems which must be resolved, this will be updated as tasks are completed and new information is found. We also have created a flexible long-term plan which outlines the key tasks that must be accomplished before the completion of our project. The long-term plan is a broad view at the project and will evolve as we make adjustments to our product. We have also scheduled two weekly team meetings to ensure a steady flow of communication and that we are staying on schedule.

Conclusion

In conclusion, the team has come up with various concepts and put in our best effort to properly plan the future endeavors taken. Including the preliminary designs of the accessible messaging application and flow-diagrams. The group has created a feasibility study as well to determine what possible roadblocks will get in the way of going through with our concepts.

• Please read the aforesaid notions if you wish to understand this document.