Meetbot: Behind the scenes

Ricard Solé Casas

July 17, 2017

Declaration

I confirm that the submitted coursework is my own work and that all material attributed to others (whether published or unpublished) has been clearly identified and fully acknowledged and referred to original sources. I agree that the College has the right to submit my work to the plagiarism detection service. TurnitinUK for originality checks.

Acknowledgements

I'd like to thank my partner Shannon for her continued support and challenges that help me grow, both professionally and personally. I would also like to thank all of you who also helped me get here.

Contents

Chapter 1

Executive Summary

This report provides a review and analysis of the SDLC project which took place June 1st-2nd, 19th-21st of 2017 at Ada National College for Digital Skills. It covers the analysis of the problem, the design process, implementation and end product, testing and validation techniques, and an evaluation of the project.

The task revolves around creating an automated chat bot for the Slack platform. This discusses why the platform of choice is Slack, the reasoning behind the technology stack choices, and how the team self-directs itself.

We achieved:

- A working bot which proves our model.
- Functioning team with a clear distribution of responsibilities.
- A collection of suggested improvement to the bot's interface.

Chapter 2

Introduction

The goal of this project was to put in practice the concepts and ideas covered in the SDLC module. A brief summary includes, but is not limited to, the following ideas:

- User Research[1]
- Agile Methodologies[2]
- Lean Startup[3]
- User stories[4]

All of these concepts are very much intertwined and do not work exclusively of each other. They are however, complementary to each other and important on their own right.

[1] https://www.wikiwand.com/en/User_Research
[2] https://www.wikiwand.com/en/Agile_software_development
[3] http://theleanstartup.com/
[4] https://www.wikiwand.com/en/User_story

Chapter 3

Analysis

Some of the organizers of the session encouraged the different teams to come up with solutions to problems they were facing in their everyday life. In our case there was a problem we encountered consistently with our colleagues and classmates. We had a hard time organizing activities with co-workers outside of work hours. Major issues encountered included: platform fragmentation, hard to track plans, attendees and their updates, no interface designed to generate plans in common environments.

Given our experience working we decided a good MVP to prove our idea would be a simple bot built on the already successful Slack model.

Chapter 4

Design

The first part of our design process was to define who our initial users would be, everything would derive from that. Our decision was to focus on *professionals who use Slack for communication at work*. It is important to highlight that we did consider other platforms, like HipChat and Telegram or Facebook's Messenger, and there is little stopping us from extending our bot to support multiple platforms. The main reason driving our choice of Slack as a pioneer platform was the ease of access to an already existing large user base on our end.

After conducting surveys, creating two personas and their corresponding user journeys, and going through user testing on mocked bots, we distilled our ideas and dreams into a small subset of features:

- Creating and deciding on a plan
- Being able to list all existing plans
- Update plans as they change

Chapter 5

Implementation

When we set out to accomplish a successful implementation to achieve maximum market value in the minimum time to market we were faced with a few decisions in how to implement our project. What is the goal of our initial users? Are we helping them and, by extension, their world if we fulfill our mission? What is our mission? What does a user's journey look like?

After thorough discussion and analysis we decided on the core features that would better the lives of our users: creating a plan, knowing what plans are available, and, last but not least, joining an existing event. These happened to be the minimal steps for our users to achieve their goal of sharing their happy hour plans with their coworkers.

What came out of this process was a functioning bot on the Slack platform which responded to real commands. UX sessions can be found in our Google Drive folder: Joe[1], Matt[2], and Danny[3].

We built a bot which responded to some basic commands that allowed users to meet IRL. Full code implementation can be found on Github[4].

[1] https://drive.google.com/file/d/0B6d4i1nhOpwKX19TYWFGTmd2V2M/view?usp=sharing
[2] https://drive.google.com/file/d/0B6d4i1nhOpwKbGVaMVlnYnY1ZWM/view?usp=sharing
[3] https://drive.google.com/file/d/0B6d4i1nhOpwKUGc0V25lcFVhSjQ/view?usp=sharing
[4] https://github.com/ada-lunchbot/lunchbot

Chapter 6

Testing

In this sequence of sessions we had two major chunks. In the first one we tried to validate our idea without a working implementation in code. The second consisted of taking all the lessons learned from the first session and have a full working bot that implemented the basic needs of our users.

In our first attempt at proving our idea a member of the team simply changed their Slack profile to mimick that of a bot. They named themselves `lunchbot`, changed their picture and statement to something more *bot-like*, and stopped participating in conversations unless summoned. That helped us gather feedback on the interface we thought we would implement to interact with the bot, along with collecting additional features that the test users considered a *nice-to-have*.

After reflecting on the feedback from the first session we implemented a bot and tested that with more precision. In the second round we recorded people's screens as we asked them to complete a journey. We noticed our concept of "technically proficient", and what that meant for interacting with our user, needed to be refined. For the recorded UX sessions please refer to links: here[1], here[2] and here[3].

[1] https://drive.google.com/file/d/0B6d4i1nhOpwKX19TYWFGTmd2V2M/view?usp=sharing
[2] https://drive.google.com/file/d/0B6d4i1nhOpwKbGVaMVlnYnY1ZWM/view?usp=sharing
[3] https://drive.google.com/file/d/0B6d4i1nhOpwKUGc0V25lcFVhSjQ/view?usp=sharing

Chapter 7

Evaluation

If I had to name one thing to take away from this experience, it could be distilled in the following sentence:

> Test early, and test often.

That is something we would've benefited from doing. We failed to test some assumptions in the beginning and it came back to haunt us in our sessions. Had this been a real commercial project with real stakeholders, the fact that our bot's discoverability and help command were unintuitive to our users would've most likely delayed launch, and cost us a lot of money in the process.

I also benefited from learning about user journeys and how to map them. That was a truly enlightening experience, and it made coming up with user stories a much more trivial task.

In terms of the coding itself, I think we did a great job at distributing the ownership of the codebase. We achieved that by having sessions of what is commonly referred to as mob programming[1].

[1] https://www.wikiwand.com/en/Mob_programming