# Abstract

# Contents Page

# Section 1: Introduction

This section provides an introduction and brief overview of the project being undertaken. The aims and motivations behind the task will be explained as well as a small evaluation of the goals and structure of this report, and how it is structured to best relay the information relevant to this projects development.

## Project Overview

### The Problem and the Motivation

The most popular meeting planning and scheduling website currently available has, amongst all its positive aspects, one noticeable flaw. Doodle is a leader in the online scheduling platform, but Professor Kaspar Althoefer had one problem with the service. He believed doodle was too rigid in its approach of meeting organisation, and presented this project to come up with a more flexible alternative to it, hence the project being originally titled ‘The Flexible Doodle’. Having similar options on the performance as well as other aspects I found could be improved upon, I chose to do tackle this issue.

Having accepted this project, I decided to take the base description and enhance it to make it a more fruitful endeavour and produce a more complete and satisfactory product at the end of the process. I was motivated to do this because I saw this as a legitimate business idea, one which if implemented correctly and efficiently could go way beyond the scope of this final year university project. It was a chance to not only demonstrate and develop my skills, but it also provided a platform to potentially create something used on a large scale. Backed up by facts and figures present later in this report, the demand and use of well implemented and functionally useful scheduling services are very high, meaning not only did undertaking this project benefit me, I also had the added motivation of knowing that I this project could provide a benefit to its users as well.

### Project Aims and Objectives

The primary aim of this project was to spend 6 months researching, designing and implementing a web based application which tackles the downfalls, and exceeds the performance of free and existing scheduling services currently available to the public. Users of the service will be inputting preferential data, which will be manipulated and presented in a manner to represent the process scheduling a meeting.

The projects defined aims and objectives are;

* Research, Test, and evaluate issues and limitations of free and existing scheduling and meeting web applications
* Design, implement and release a free web application which tackles most or all of the identified issues with the existing scheduling solutions
* Rigorously test the implemented solution with samples of existing target audience members to ensure that the provided service adequately satisfies the targets requirements.

The projects sub-aims and objectives are;

* Create an efficient and ergonomically designed interface allowing users to efficiently create and take part in meeting organisation and scheduling
* Design an aesthetically pleasing interface for the web application so users are drawn into and satisfied while using the service.
* Ensure the provided application is safe and secure to use while not sacrificing any potential beneficial usability features.
* Implement the web application in a manner considering longevity and capability with minimal maintenance and attention required to ensure the service is long lasting and efficient.

### Challenges

As with any project, there are challenges faced during the process of completion. The main challenge in attempting this project was going up against the dominance in the already existing services out there. Doodle and other competing companies have been in the field for years, giving them a lot more time to grow, test, and develop, meaning the quality of the output of this project needed to be that much higher.

Another challenge was learning and developing in relatively new and advanced languages and techniques such as Node Js and JQuery, which is required to ensure a certain bench mark for complexity and depth to the project.

## Report Overview

The goal of this report is to give an insight into the process of this project from start to finish. Taking a look at the inner workings of every phase of development, as well as considering and assessing multiple aspects in order to come up with a critical and accurate evaluation of my performance, and the success of this project. This report is structures to represent the flow of the product life cycle through development.

# Section 2: Background Research and Literature review

This section provide a detailed analysis of the research undertaken in order to gain a stable platform and idea of how to start the project. It includes a brief but beneficial analysis of existing solutions to the problem by competitors, as well as an insight into concept and principals used to develop the product.

## 2.1 Existing Solutions

In order to get a better understanding of which features to include in my web application to the scheduling problem, I had to research and analyse existing solutions. This allowed me to see the existing flaws and decide which to tackle, as well as highlight possible points to include to make my solution better and more efficient than the current ones.

### 2.1.1 Doodle

#### 2.1.1.1 Overview

Doodle is the leading provider for free online scheduling, and provides a quick and easy form of meeting organising based on a binary check box availability model, where users choose yes or no for a given options. According to doodle themselves; (Doodle: easy scheduling, 2017) **“**Doodle is an online scheduling tool that can be used quickly and easily to find a date and time to meet with multiple people.  First you suggest dates and times for your event participants to choose from, then Doodle creates a polling calendar that can be sent to participants for feedback.  As each participant selects the dates and times from the polling calendar that he or she is free, Doodle aggregates the responses and tells you which option works best for everyone.

Getting everyone on the same page is hard enough with just five other co-workers, let alone fifty new volunteers you may have never met.  Doodle can be a valuable tool for coordinating a large event like volunteer training as well as a small event like a monthly meeting with the board.”

Doodle is by far the most used and most popular online scheduling platform, due to its simple and attractive interface and useful features. Statistics compiled by doodle themselves (Eberle, 2018) show that;

* Doodle is used by more than 20 million users per month worldwide
* Over 17 million polls were created by users of doodle in 2013
* A doodle poll is made on average every 2 seconds

#### 2.1.1.2 Positives

Doodle is a leading member of the free scheduling provider platform and has many positive aspects. Some of these include;

* A simple and straight forward process to create and participate in a meeting
* Attractive and relaxed user interface and design
* Large number of useful features and functions, including calendar integration and a user dashboard

#### 2.1.1.3 Negatives

In order to come up with something better than the existing solutions, the negative aspects must also be identified. Some negative aspects identified were;

* Rigid approach to representing availability. Time slots are pre-defined and no overlap or layered preference is applicable. Everything is binary, either ‘I can’ or ‘I can’t’.
* Just presents user info, and actually doesn’t do much more with the data. The decision is left to the user in the end, with no help from the system.

### 2.1.2 NeedToMeet

#### 2.1.2.1 Overview

Need to Meet is another leading scheduling service provider, however it is different in its approach to Doodle. Need to Meet has a more professional looking theme, and works as a support application, integrating around a user’s calendar rather than manual input of times.

(Needtomeet.com, 2017) **“**NeedToMeet is a global provider of meeting scheduling software that allows individuals and businesses to find a time that works for everyone. The product streamlines the process of [meeting administration](http://www.needtomeet.com/dashboard), [attendee polling](http://www.needtomeet.com/easy-polling), and scheduling any type of meeting via both online and mobile interfaces, while removing the hassles of lengthy email chains, time zone conflicts, and phone tag.

It’s powerful administrative features allow a meeting organizer to manage the details of all their meetings in a single dashboard. NeedToMeet can be used with any browser and can be installed as a Microsoft Outlook add-in to fully capitalize on the Exchange environment.”

#### 2.1.2.2 Positives

* Very well integration with outlook calendar making the process of inputting availability a lot easier for the user
* Good user integration, making it easy for users to collaborate and communicate to come up with the best time to meet.

#### 2.1.2.3 Negatives

* Not the most pleasing aesthetically, with the use of a lot of tick boxes and menus to navigate around options once the preferences have been set, especially on the mobile application.

### 2.1.3 Evaluation of Competitors

Both these existing solutions tackle the scheduling problem in different manners due to the user marker they are targeting. Doodle is a mass market product, with anyone able to use the product, hence the non-account usable functionality and the quirky and bright visuals. Whereas NeedToMeet is more professional website tailored for the business meeting.

Overall both perform very well due to their long standing existence in the market, but both have flaws which can definitely be tackled for the final outcome of my project.

## 2.2 Logic

The logic of a program is the computational characteristic and classification that the program uses to perform or reason during computation.

### 2.2.1 Fuzzy Logic Over Traditional Logic

Fuzzy logic is a relatively new logical method of thinking and computation. It strays from the original traditional logic of definitive Boolean classification, where something is either true or false (0 or 1). Fuzzy logic operates in the grey are between these choices and allows for a more generalised standard of logic. Fuzziness was intended for the classification of inherently vague concepts such as tallness, e.g. ‘Person x is tall’ would have a degree of truth 0.8.

Fuzzy logic was originally developed by Dr. Lotfi Zahed of the University of California at Berkley University in the 1960s. Dr. Zadeh realised that natural language was something which couldn’t always be categorised into absolute terms of 0 and 1 while he was researching the problem of how computers understand patternss of natural language (WhatIs.com, 2017)

Fuzzy logic aims to work similar to the way the human brain computes and acknowledges states. Rather than having definitive classifications, aspects are assigned their own ‘degrees of truth’. 0 or 1 are only assigned to extreme cases, whereas the majority of classifications are assigned a value between this range.

### 2.2.2 Suitability of Fuzzy Logic

The main purpose of this project is to take the personal preference and availability entered by the users. Then come up with and present the most ideal meeting intersect for the multiple attendees. These two things aren’t things that can be assigned a definite Boolean value. A user can be available for a period of 2 hours however would prefer if the meeting could be held in the first hour rather than the second.

Using fuzzy logic would allow the user to define their preferences and assign higher priorities to the choices they have selected. Fuzzy logic is the logical approach which allows this project to tackle the main issues with the already existing issues that his project hopes to solve. The project will implement fuzzy logic in its implementation, allowing the user to specify preferences among their availability choices, to be considered when making the final decision.

## 2.3 Design Principals

## 2.4 Proposed System

The existing options for meeting schedule websites are very popular and affective, but one major down fall across the sector is the rigidness of usability. Doodle being the leading option only allows users to apply Boolean values to pre-set times and dates, limiting the amount of control the users have. For example; the admin may set meeting times to start on the hour every hour i.e. 10am, 10am, 12pm etc. However, I as an invitee am available for an hour but only free from 10.20am to 11.20am. There is no method of specifying this preference. This is something my solution will attempt to tackle.

My system will allow the users to pick a specific time (by the minute) in a stated range which adds flexibility to choices i.e. 10.34am. However, on top of this, fuzzy logic will be incorporated to the implementation so that users can assign preferences to their choices. For example; I could be free from 11am to 2pm for an hour meeting, however would prefer if the meeting could occur in the hour between 12.35pm and 1.35pm. Therefore, I would assign a higher preference to this hour range. This would then be held and considered in higher regard when calculating the best meeting time.

## 2.5 Web Application Development

All web applications have three components which need to be carefully chosen based on the behaviour and requirements of the application. The components are;

### 2.5.1 Client Side Scripting

The client is at the end of the product using the system on a browser, which is the location of client side technology is unpacked and processed. The client is wherever the users are viewing the application including; mobile device, laptop or Desktop computer.

Client side scripts are embedded inside the web apps HTML code which is held on the server in a language of the developer’s choice. The browser downloads the code off the server and processes the instructions, at which point is presented to the user on the browser. The user can then interact with the app, an on occurrence of events, requests are sent back to the server and carried out.

#### 2.5.1.1 HTML

HTML (Hypertext Mark-up Language) is the base coding language most popularly used for website development - *(Hyperlink - works with the concept of using hyperlinks to navigate around and use the functions of a website. Mark-up – marking the contents within the HTML tags in code*). HTML consists of a series of short codes written in ‘tags’ which allow the user to display information on a web browser, like images and text. HTML codes are compiled and rendered by web browsers where tags are converted to their coded renderings and displayed on the web browser.

#### 2.5.1.2 JavaScript

JavaScript is a complicated programming language which is added on top of the base HTML code to increase the features and functionality of a webpage from a front-end point of view. JavaScript allows for a more dynamic webpage to be created, where information is required to do more than just sit static on a page upon render. It allows dynamically updating content.

#### 2.5.1.3 CSS

CSS (Cascading Style Sheet) is a simple coding language with the role of altering the aesthetics of a webpage. It has no effect on functionality, it is purely processing the visuals of an application. A CSS file is a simple file with a series of ‘rules’ detailing the visual appearance of a particular tag in the webpage, linked via a class declaration, for example, making a button green, or deciding spacing between features etc.

#### 2.5.1.4 Bootstrap

Bootstrap is a free to use open source front-end coding framework which is used with front end coding languages such as HTML and JavaScript to produce a responsive and cross platform suitable web application. It is broken down into 3 main files;

* bootstrap.css – a CSS framework
* bootstrap.js – a JavaScript/JQuery Framework
* glyphicons – an icon font set

The benefits of using a framework such as bootstrap are;

* reduces repetitive code
* makes use of adaptive content allowing app usage across multiple platforms
* increases consistency in aesthetics of a web app
* allows for quick and efficient prototyping and designing
* ensures cross browser compatibility

### 2.5.2 Server Side Scripting

The server is a remote location which can be located anywhere in the world, and is responsible for housing data and running the web applications back end architecture, processing requests, and sending pages to the web browser.

#### 2.5.2.1 Node.js

Node.js is a server-side platform built on Google Chrome’s JavaScript Engine (V8 Engine). It is a free and open source server side framework based on the JavaScript coding language. The definition according to official Node documentation is a platform built for fast and scalable network application. Node.js uses event-driven, non-blocking I/O models that make it lightweight and efficient, perfect for data-intensive real-time applications that run across distributed devices.

***Features of Node.js:***

* *Asynchronous and event driven*

Every APIs in the Node.js Library are asynchronous and non-blocking. This means that Processes never wait for the return of data from an API call before moving to the next one. The server moves to the next API as soon as soon as one is called.

* *Very Fast*

Extremely quick code execution due to being built on Google Chrome’s V8 JavaScript engine.

* *Single threaded but highly scalable*

Uses a single thread model with even looping. Event mechanisms aid the server perform in a non-blocking way and makes the server highly scalable. Node.js uses a single threaded program and the same program can provide service to a much larger number of requests than traditional servers like Apache HTTP Server

* *No buffering*

Node.js applications never buffer any data. These applications simply output the data in chunks

#### 2.5.2.2 Express.js

Express.js is the most popular Node web framework, and is actually the underlying library for a number of other popular Node web frameworks. Express is used alongside Node and allows implementation of mechanisms such as;

* Request handlers for different HTTP verbs (GET, POST etc.) at different URL paths (routes)
* Integration with ‘views’ in order to generate responses by inserting data into templates
* Set common web application settings like the port of use and the location of templates that are used for rendering responses.
* Add additional request middleware at any point within the pipeline.

### 2.5.3 Database Management System (DBMS)

A DBMS is a system software for creating and managing databases. It provides users and developers with the ability to create, retrieve, update and manage data through the use of special semantics. A DBMS makes it possible for an end user to create, read, update and delete data in a database. The DBMS manages three important things; the data, the database engine and the database schema. These three things help provide concurrency, security, integrity, and uniformity.

***Advantages of using a DBMS:***

* Allows both developers and end users to access the same data while still maintaining data integrity
* Data is better protected and maintained when shared using a DBMS rather than creating new iterations of the same datasets
* Provides a locking mechanism for concurrent access
* Efficient handling to balance needs to multiple applications accessing the same data blocks
* Swift recovery from errors and data loss
* Simple access and usability

#### 2.5.3.1 MySQL

MySQL is an Open Source SQL database management system, developed, distributed and supported by Oracle Corporation. The SQL part of ‘MySQL’ stands for “Structured Query Language”, which is the most common standardised language used to access databases.

***Features of MySQL:***

* *MySQL is very fast, reliable, scalable and easy to implement*

MySQL can comfortably work across all platforms, and along other applications, requiring little to no attention. Was originally developed to handle large amounts of data quicker than existing solutions and has been successful up to this point.

* *MySQL server works with client/server or embedded systems*

MySQL database software is a client/server system that consists of multi-threaded SQL servers that support several back and front-end libraries and tools.

* *Largely available supporting software*

Developed in close cooperation with users therefore it is very easy to find useful and satisfactory supporting software for the application needs.

#### 2.5.3.2 MongoDB

MongoDB is an open source database that uses document-oriented models. It is a relatively new database type, invented in the mid 200s and operates under the banner of NoSQL. It works with an architecture of collections and documents, consisting of sets of key-value pairs.

***Features of MongoDB:*** (Home and Database, 2017)

* *MongoDB supports dynamic schema design*

Allows the documents in a collection to have different fields and structures

* *Allows you to Store large volumes of data that often have little to no structure*

A NoSQL database sets no limits, and allows you to add different types of data as your needs change.

* *Make the most of cloud computing and storage*

MongoDB can load a high volume of data and give you lots of flexibility and availability in a cloud-based environment, with built-in sharding solutions that make it easy to partition and spread out data across multiple servers.

* *Develop and release quickly*

With MongoDB’s dynamic schemas, data doesn’t need to be prepped ahead of time, and your team can incorporate anything new, quickly, and at a lower cost.

* *Scale database architecture efficiently and inexpensively*

Easy to spread data out across commodity hardware on-site or in the cloud without needing additional software.

#### 2.5.3.3 SQLite

#### 2.5.3.4 DBMS Conclusion

After analysing the features of the two researched DBMS available to me, I have decided to use MySQL for the management and implementation of my project. Although MongoDB without a doubt has a lot of benefits, in terms of implementation I have more experience with MySQL therefore it is safer to opt for MySQL.

# Section 3: Methodology

## 3.1 Requirement Gathering Methodology

## 3.2 Software Development Methodology

A software development methodology in software development is a framework that is used to structure, plan and control the process of developing a system (Itinfo.am, 2017). These different methodologies have distinctly and in some cases minute differences which affect the progress of development. Based on the nature of a project and its requiems, a specific methodology is used. After conducting some research, the most suitable methodologies for me are:

### 3.2.1 Waterfall Model:

Considered as the traditional and most common. Attempts development with a simplistic linear progressive flow with a specified sequence of steps carried out in order to achieve the final product. Each progressive level relies on and only starts upon completion of the previous step, with no reverse transition in the progress of the project.

***Pros*:**

* Simple to implement and apply
* Time efficient
* Allows for easy testing and analysis
* Allows strong managerial capabilities by applying deadlines and specific tasks for development

***Cons*:**

* Can only be used if the software requirements are specific and known
* Strict flow means no overflow in stages, i.e. no programming while testing
* Outcome of project is only known at the end of the as there is no intermediary testing
* Not suitable for long or ongoing projects
* Not suitable for handling unforeseen problems or changes for the project

### 3.2.2 Agile Model

A more complicated and extensive development methodology. Works on the concept of having short period of development iterations with a flexible flow between levels. Each iteration is planned as a small software project in its self, creating something in its self-deliverable. With the project being re-evaluated at the end of every iteration.

***Pros:***

* Well suited to dealing with unforeseen problems and changes
* Improved final outcome by dealing with all defects as they appear
* Target market can have early exposure and give feedback

***Cons:***

* High focus on implementation therefore other parts such as documentation and planning may suffer
* Chance of losing track of progress as the goals aren’t clear in terms of iterations

### 3.2.3 Spiral Model

A complicated development model with the goal of reducing risks at the early stages of production. A spiral model starts the project on a small scale, focusing on a singular aspect, explores the risks in this aspect, makes a plan to deal with the risks, and then decides whether or not the next step of the process should be taken. This works in a spiral form working outwards increasing the aspect every step.

***Pros:***

* Considerably reduces the risk factor of the project
* Great for large and complicated projects
* Is very flexible in terms of extra functionality
* Suitable for high risk projects

***Cons:***

* Financially inefficient
* Failure to spot and deal with risks may damage entire project
* not appropriate for low risk projects
* could hold back progression

### 3.2.4 Software Development Methodology Conclusion

After critically comparing the 3 models described above and comparing the pros and cons with the nature of my project, I have decided that the Agile software development model would be best suited for my needs.

The project I am working on works with the idea of modules. Each function of the website has its own behaviours and requirements. They all have specific focus and are brought together in the end to complete the whole application. Because of this I feel that having an agile model is suitable as it gives the greatest flexibility and allows for the most efficient progression in terms of overall progress.

The ability to make changes at any stage and modify and analyse at the end of every step is also a major aspect which this project could benefit from. Web development, especially front end and database coding has a lot of problems if not careful, and the flexibility in development is well suited for this project.

Having the ability to also continuously test the outcome and quality of the project at the end of every iteration also increases the chances of making an efficient and working website. The agile model is more user focused and this project is mainly a consumer service so user focus is of paramount importance to the success of this project.

## 3.3 Database Design Methodology

This project relies heavily on the storage, access and manipulation of data from a centralised online location. Therefore, having a well designed and implemented database is of upmost importance to the success of this project. In order to do so, core database design methodologies must be undertaken. The phases of database design methodology are;

### 3.3.1 Conceptual Database Design (Conceptual Schema)

Begins with the creation of a conceptual model of the system/enterprise. It is a high-level description of valuable information for the system. It maps entities belonging to the system and the relationships between them to see how they interact and rely on each other. At this phase, specific features such as programming languages, hardware and performance issues.

### 3.3.2 Logical Database Design

A logical database design takes the conceptual schema one step forward and includes the definition of types of information needed within the database. It is a method of constructing a model to represent the specific information used by the system. The logical database design maps the conceptual model to a logical model, which is in turn used as a source of information for the physical data model. The output of this process is a global logical data model consisting of an Entity- Relationship diagram, relational schema, and supporting documentation that describes this model

### 3.3.3 Physical Database Design

Is a representation of the implementation of the database. It describes the base relations, file operations and indexes used to achieve efficient data storage and access and any associated integrity and security measures. Logical model is concerned with what is stored, where as a physical model considers more how it is stored.

## 3.4 Software Testing Methodology

Software testing is a vital part of the development process, and much like development there are different types of testing methods, each with their own characteristics. Testing usually occurs nearer the end of the development cycle but can be carried out at any stage once implementation has begun while using the Agile development model being used for this project. The goal of testing is to identify and fix errors and bugs in the final product and is a method of quality assurance. The main aspects of software testing are;

### 3.4.1 Functional testing

Is the process of comparing the software against its requirements. This is done by comparing it to the points included in the functional specification for the project. Functional testing is typically broken down into 4 parts;

#### 3.4.1.1 Unit Testing

This involves testing each individual module or component of the application. Each function of the specified module is tested by a specialised unit tester written in the same language as the program and faults or bugs are identified and handled. In many cases the tests are created before the module, as it ensures the end product meets the original criteria it was based upon.

#### 3.4.1.2 Integration Testing

This build on from the unit tests, where individual modules are tested, by integrating successful unit tested modules with each other and then tested for faults. Even though an individual module may run smoothly and correctly, upon integration there could be unforeseen clashes and errors produced. This method is usually carried out by interfacing hardware and software from the entire system and then testing for errors.

#### 3.4.1.3 System Testing

Takes the entire, complete and fully integrated system and then checks for bugs and errors. Takes modules which are completely integrated and have passed both unit and integration tests to see if the entire system is complete and error free.

#### 3.4.1.4 Acceptance testing

Is the final phase of the functional testing process. It is carried out to ensure that functionality wise, the delivered software complies completely and effectively with all the functional requirements stated in the specification. This can be done in one of two ways (or both);

* internal acceptance testing
  + Test is carried out by a member of the development team
* external acceptance testing
  + test is carried out by a member of the consumer group

### 3.4.2 Non-functional Testing

Opposite to functional testing, non-functional testing is the process of comparing the product to the non-functional requirements highlighted in the original specification. This typically involves the software being tested against measurable defined qualities. The stages of non-functional testing are;

#### 3.4.2.1 Performance, Load and Stress testing:

Measuring the performance of the software under expected and possible strenuous usage tasks, such as simulating with large number of users or data.

Load testing is testing to see how the performance is affected when the system is subjected to the expected level of load usage.

Stress testing is testing to identify the point of failure. Increase load used in test until software crashed or is incorrect.

#### 3.4.2.2 Security and Vulnerability Testing:

The process of testing the software for confidentiality, integrity, authentication, availability and non-repudiation. Tests are conducted internally and individually to prevent unauthorised access to the software’s code.

The approach has evolved considerably in recent times. Originally security testing was something that was dealt with at the latter stage of development however the increased occurrence of cyber-attacks, security checks and precautions are implemented in parallel with development from the start.

#### 3.4.2.3 Usability Testing:

Tests the experience of the end user for the software. Tests and analyses the ease in which the end users can understand how to use and navigate the software and how much training is needed in order to reach a suitable level of knowledge to effectively use the software. The main aspects that usability testing focuses on are; learnability, efficiency, satisfaction, memorability and errors.

#### 3.4.2.4 Acceptance Testing:

Similar to functional testing, this phase tests how accurately the end product satisfies the originally stated non-functional requirements in the specification. A major part of this phase is testing to see if the software is compatible with all specified operating systems, hardware platforms, web browsers, mobile devices, and other third-party designed platforms.

# Section 4: Requirements and Analysis

# Section 5: Design

# Section 6: Implementation

## 6.\_\_\_Optimal Time Algorithm and Functionality:

import operator

l = [(1, 3), (2, 8), (14, 22), (4, 15)]

lower\_bounds = []

upper\_bounds = []

for pair in l:

lower\_bounds.append(pair[0])

upper\_bounds.append(pair[1])

count\_table\_lower = {}

count\_table\_upper = {}

def intoTable(li, dic):

global l

for i in li:

for p in l:

if i >= p[0] and i <= p[1]:

if i in dic:

dic[i] = dic[i] + 1

else:

dic[i] = 1

return dic

count\_table\_upper = intoTable(upper\_bounds, count\_table\_upper)

count\_table\_lower = intoTable(lower\_bounds, count\_table\_lower)

x = max(count\_table\_lower.items(), key=operator.itemgetter(1))

y = max(count\_table\_upper.items(), key=operator.itemgetter(1))

# print(str(count\_table\_upper))

# print(str(count\_table\_lower))

print((x[0], y[0]))

O(n log n)

# Section 7: Testing

# Section 8: Conclusion

# Section 9: Future Work

# Bibliography

Itinfo.am. (2017). *Software Development Methodologies*. [online] Available at: http://www.itinfo.am/eng/software-development-methodologies/ [Accessed 29 Oct. 2017].

blogs, V. (2017). *12 Best Software Development Methodologies with Pros & Cons*. [online] Web Solutions Blog. Available at: http://acodez.in/12-best-software-development-methodologies-pros-cons/ [Accessed 1 Nov. 2017].

Segue Technologies. (2017). *Waterfall vs. Agile: Which Methodology is Right for Your Project?*. [online] Available at: https://www.seguetech.com/waterfall-vs-agile-methodology/ [Accessed 2 Nov. 2017].

Ecomputernotes.com. (2017). *What is Database Design Methodology? Different Phases of Design Methodology.*. [online] Available at: http://ecomputernotes.com/database-system/rdbms/phases-of-design-methodology [Accessed 5 Nov. 2017].

Docs.oracle.com. (2017). *Overview of Logical Design*. [online] Available at: https://docs.oracle.com/cd/A81042\_01/DOC/server.816/a76994/logical.htm [Accessed 6 Nov. 2017].

Scientific American. (2017). *What is 'fuzzy logic'? Are there computers thatare inherently fuzzy and do not apply the usual binary logic?*. [online] Available at: https://www.scientificamerican.com/article/what-is-fuzzy-logic-are-t/ [Accessed 10 Nov. 2017].

WhatIs.com. (2017). *What is fuzzy logic? - Definition from WhatIs.com*. [online] Available at: http://whatis.techtarget.com/definition/fuzzy-logic [Accessed 11 Nov. 2017].

Foundation, N. (2017). *About | Node.js*. [online] Node.js. Available at: https://nodejs.org/en/about/ [Accessed 8 Dec. 2017].

Dev.mysql.com. (2017). *MySQL :: MySQL 5.7 Reference Manual :: 1.3.1 What is MySQL?*. [online] Available at: https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html [Accessed 6 Nov. 2017].

Home, H. and Database, S. (2017). *What is MongoDB and Should You Use It?*. [online] Hiring | Upwork. Available at: https://www.upwork.com/hiring/data/should-you-use-mongodb-a-look-at-the-leading-nosql-database/ [Accessed 13 Nov. 2017].

Doodle: easy scheduling. (2017). *What is Doodle and how does it work: an ...*. [online] Available at: https://help.doodle.com/customer/portal/articles/761313-what-is-doodle-and-how-does-it-work-an-introduction [Accessed 8 Dec. 2017].

Eberle, T. (2018). *Numbers | Doodle Blog*. [online] En.blog.doodle.com. Available at: https://en.blog.doodle.com/category/numbers/ [Accessed 30 Nov. 2017].

Needtomeet.com. (2017). *About NeedToMeet*. [online] Available at: http://www.needtomeet.com/about-us [Accessed 8 Dec. 2017].

GmbH, a. (2017). *MAMP & MAMP PRO*. [online] Mamp.info. Available at: https://www.mamp.info/en/#controlcenter [Accessed 8 Dec. 2017].

Sequelpro.com. (2017). *Sequel Pro*. [online] Available at: https://www.sequelpro.com/ [Accessed 8 Dec. 2017].

**In-text:**(Lhomme, Gotlieb and Rueher, 1998)

Lhomme, O., Gotlieb, A. and Rueher, M. (1998). Dynamic optimization of interval narrowing

algorithms. *The Journal of Logic Programming*, 37(1-3), pp.165-183.

Chen Kai. (2010). Design Principle for Ubiquitous Computing

**In-text:**(Norman, 2013)

Norman, D. (2013). *The design of everyday things*. New York: Basic Books.

# Appendix