CLOUD DIAL

A cloud based visual bookmarking service

FUNCTIONAL SPECIFICATION
Shane McQuillan - 58600724

Table of Contents

1. Introduction

1.1 Overview

The goal of this project is to develop a **cloud** based **visual bookmark**ing system, interlinked with Facebook. Cloud Dial will be a **web application** that allows users to store, manage, retrieve and share their bookmarks anywhere, any time, from within their web browser.

This system aims to be central to a users' web browsing. It should be the first thing they see when they open their browser, a new tab, or new window. Doing so they can quickly visit their favourite sites, and collected bookmarks. In Cloud Dial each bookmark will be identifiable by a unique image. This will make it visually appealing and extremely practical to use in this way.

Identifying bookmarks further will be made possible by means of **textual tags, content summarisation**, and categorisation into groups and sub-groups; all of which will be suggested by Cloud Dial. This rich variety of information will ensure efficient browsing, and effective bookmark retrieval.

It is hoped this system will be used by all web browser users, but more specifically, by those who bookmark regularly. An ideal user bookmarks on several different devices, bookmarking pages of various topics. For people who bookmark a lot, and need to retrieve specific bookmarks on many devices, Cloud Dial is the perfect solution.

Bookmarking is a standard feature of most web browsers, and while Cloud Dial may offer additional functionalities; it is still important it performs as closely to the standard model as possible. It will undoubtedly be challenging, but if it fails to do so, there is the risk of harming the user experience. This would be detrimental to the product's future.

There is much scope for expansion of this system. As such it will be designed with further additions and improvements in mind.

1.2 Business Context

In 2010 over twenty billion dollars was spent on online advertising, and this is expected to increase year on year. Cloud Dial has the potential to be an excellent platform for advertisement. Some possibilities include -

• Pre-existing bookmarks: When a user creates an account their bookmarks are pre-populated with business sponsored bookmarks.

• Bookmark suggestion: Cloud Dial intends on suggesting bookmarks to users based on their preferences. Some could be business sponsored.

• Additional bookmarks: Advertisers' bookmarks could be displayed along with a users' bookmarks collection.

• Sales of anonymous data: Statistics built up on users could be sold to advertisement companies. These statistics would be anonymous to avoid privacy concerns.

1.3 Glossary

Cloud: A loosely defined term for any system providing access via the Internet to processing power, storage, software or other computing services, often via a web browser.

Visual bookmark: A direct link to a predefined web page represented by an image.

Web application: A website that behaves like software.

Textual tags: Freely chosen keywords used to categorise data. Often used for the purpose of future retrieval of the data in question.

Content summarisation: A shortened version of specific content, focusing on its main points.

Bookmarklet: A bookmarklet is a small Javascript application stored as the URL of a bookmark in a web browser, or as a hyperlink on a web page.

Bookmark identifying options: These are items used to individualise bookmarks - Textual tags, images and content summarisation.

Amazon Elastic Cloud Compute: A system by Amazon offering scalable, pay-as-you-go compute capacity in the cloud.

HTML5: HTML is the language for describing the structure of Web pages. HTML5 is currently under development as the next major revision of the HTML standard.

MVC: Model-View-Controller, an architecture for building interactive applications.

Javascript: JavaScript is the most popular scripting language on the internet, and works in all major browsers.

Backbone.js: Backbone supplies structure to JavaScript-heavy applications.

jQuery: jQuery is a fast and concise JavaScript Library that simplifies HTML document traversing, event handling, animating, and Ajax interactions for rapid web development.

Node.js: Node.js is an event-driven I/O server-side JavaScript environment. It is intended for writing scalable network programs such as web servers.

MongoDB: MongoDB is an open source, scalable, high-performance, schema-free, document-oriented database.

2. General Description

2.1 Product / System Functions

Cloud Dial will have two sets of different functionalities, common and advanced. Common functionalities will be available on all devices. Advanced functionalities will be available for desktop and tablet users only, for extra control over their Cloud Dial account. These features will not be available on mobile devices in order to avoid the interface becoming overly complex.

Listed below are the two sets, and the functionalities in each -

2.1.1 Common functionalites

•   Add bookmarks.

•   Search bookmarks.

•   Automatically share additions on Facebook.

•   Suggest bookmark identifying options (tags, summary, images etc.)

•   Privatise bookmarks.

2.1.2 Advanced functionalities

•   Suggest bookmarks (desktop only).

•   Create groups.

2.2 User Characteristics and Objectives

There are many different types of Cloud Dial user. Cloud Dial focuses on satisfying two types - Those who bookmark occasionally, but visit the same web pages regularly (basic user), and those who bookmark regularly, and often need to retrieve specific bookmarks on many devices (power user).

2.2.1 Basic user

This user will benefit from Cloud Dial's visual element as well as it's ability to categorise. Often visited pages could be stored in an accessible group, represented by a visual appealing and relevant image. Using this as their home page such a user could quickly visit these sites, greatly increasing productivity.
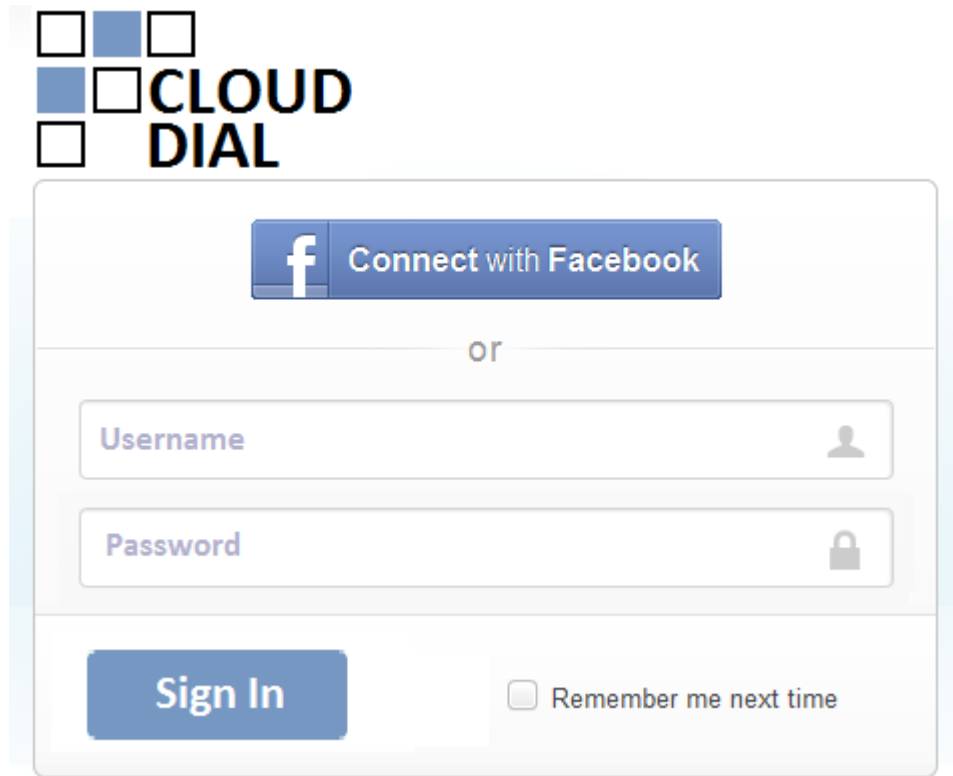
2.2.2 Power User

This user will benefit from Cloud Dial's ability to individualise bookmarks through multiple identification options. This, along with the visual element; will make retrieving specific bookmarks more straightforward on all devices.

2.3 Operational Scenarios

Below are the main operational scenarios of Cloud Dial, along with the main screen shot of each. The screen shots are mock-ups of the proposed system, and are not intended as the final user interface. They are used primarily for illustration purposes.

2.3.1 Logging in

Using Cloud Dial -

A registered user visits Cloud Dial. At the login screen they provide their username and password, and submit. Once validated they are presented with their account page.

Using Facebook -

A Facebook user visits Cloud Dial. They have not registered with Cloud Dial, so decide to log in with their Facebook account. They provide their account information and submit. They then give Cloud Dial permission to use their Facebook account (first time), after which they are finally presented with their account page.

2.3.2 Signing up

A user visits Cloud Dial for the first time. They don't have a Facebook account, so they create a Cloud Dial specific account. They provide a username, password and email address. Once submitted they confirm the account using the link provided in a sent confirmation mail.

2.3.3 Adding a bookmark

Bookmarklet

Bookmark image

Bookmark location

Textual tags

Content Summary

Account addition option

Privacy option
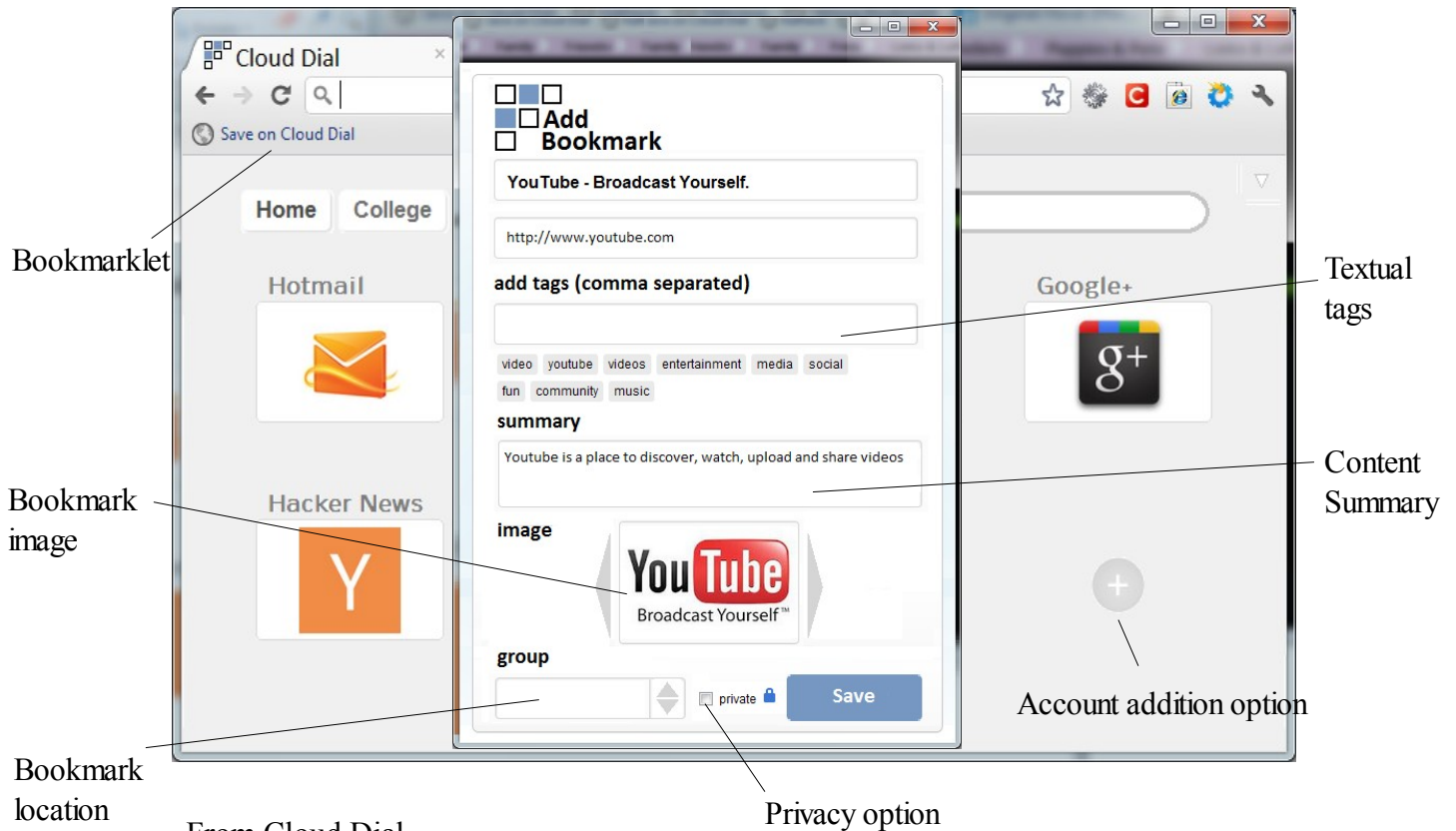
From Cloud Dial -

A registered user decides to bookmark a web page. They select the add option on their Cloud Dial account, upon which the bookmark addition screen appears. The user then provides the URL of the desired bookmark. Cloud Dial analyses the URL and provides **bookmark identifying options**. The user chooses textual tags to describe its content, an image to represent it, whether it's public or private, some summary text and what group it belongs too. They then save it.

Using Cloud Dial bookmarklet -

A logged in user decides to bookmark a web page. From the desired web page they select the Cloud Dial bookmarklet. The bookmark addition screen appears with the URL of the desired bookmark pre-populated, and the bookmark identifying options provided. The user chooses tags to describe its content, an image to represent it, whether it's public or private, some summary text and what group it belongs to. They then save it.

2.3.4 Searching for a bookmark
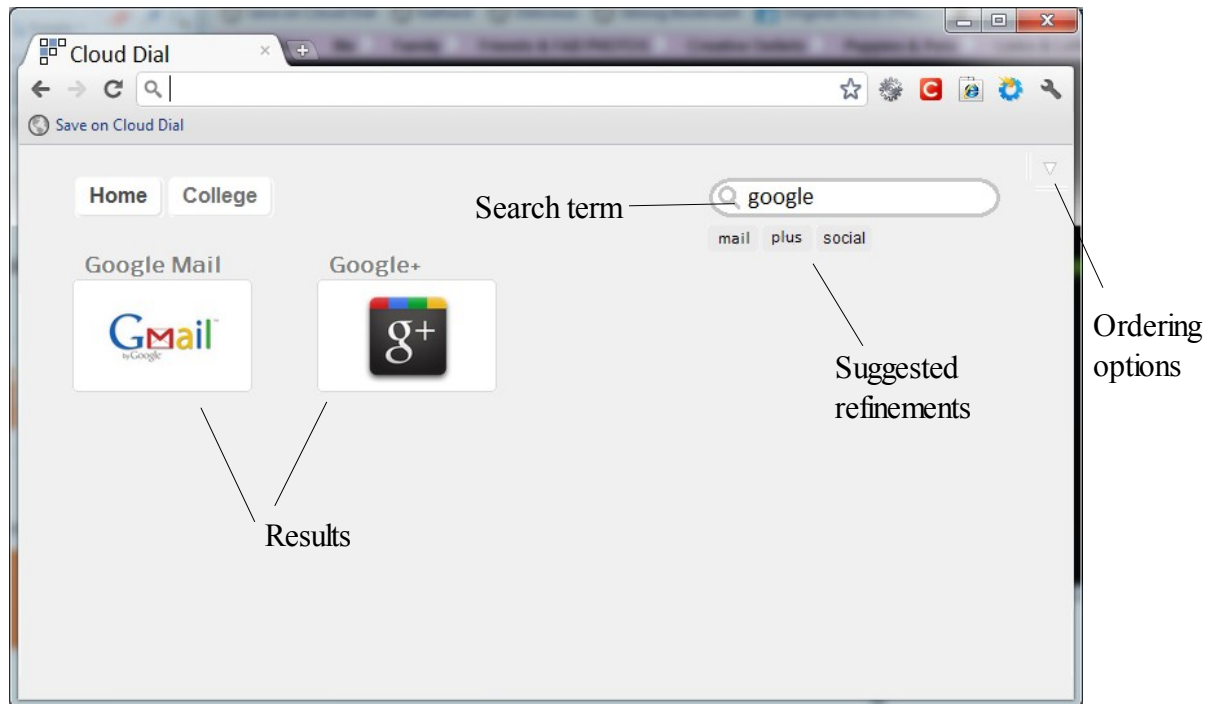
A registered user wishes to visit one of their many bookmarks. It is quite a while since they added this bookmark, and they cannot remember how they described it. They know it's related to flowers, so they enter they keyword "flowers". This user has bookmarked many flower related pages, and though this narrows the search field, there is still numerous bookmarks to choose from.

Cloud Dial suggests several keywords to refine the search which triggers the users memory. They select another keyword they believe to be related to this bookmark. This decreases the search space considerably, but not completely.

The user knows they added this page a considerable time ago. They decide to order the remaining selection in reverse chronological order. The desired bookmark is at the top of the list, and the search is successful.

2.3.5 Sharing on Facebook:

A Facebook user logins into Cloud Dial using their Facebook account. While giving Cloud Dial permission to use their Facebook account, they also have the option of allowing Cloud Dial to update their Facebook account on addition of bookmarks. The user chooses to allow this.

The user later adds a publicly visible bookmark to their Cloud Dial account. Their Facebook account is immediately updated to reflect this addition.

2.3.6 Creating groups
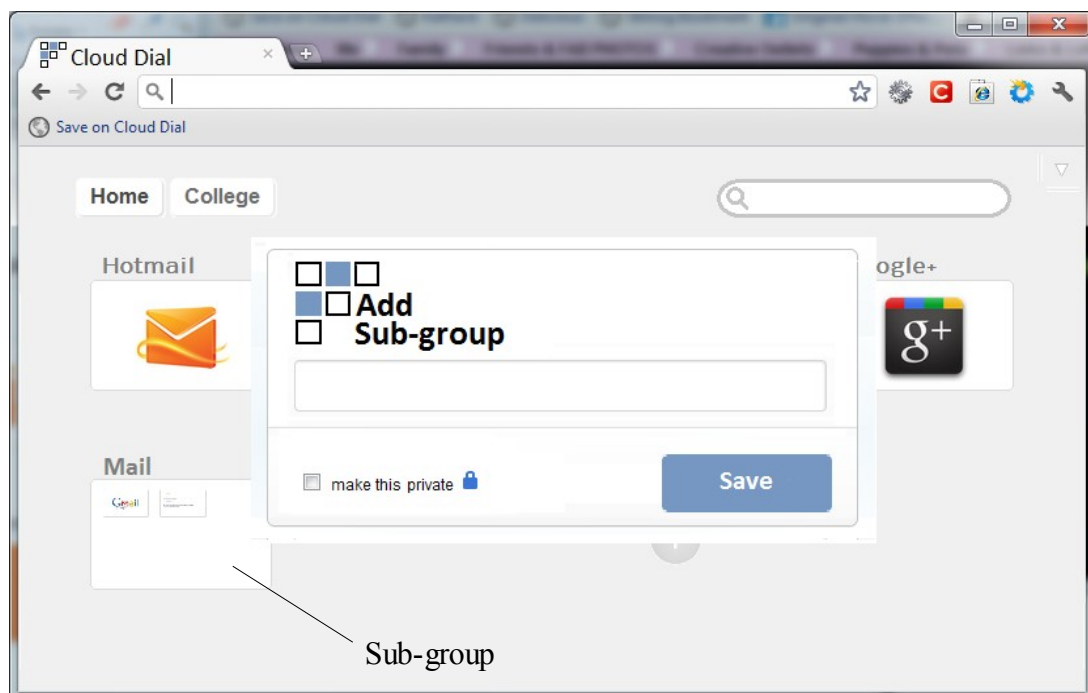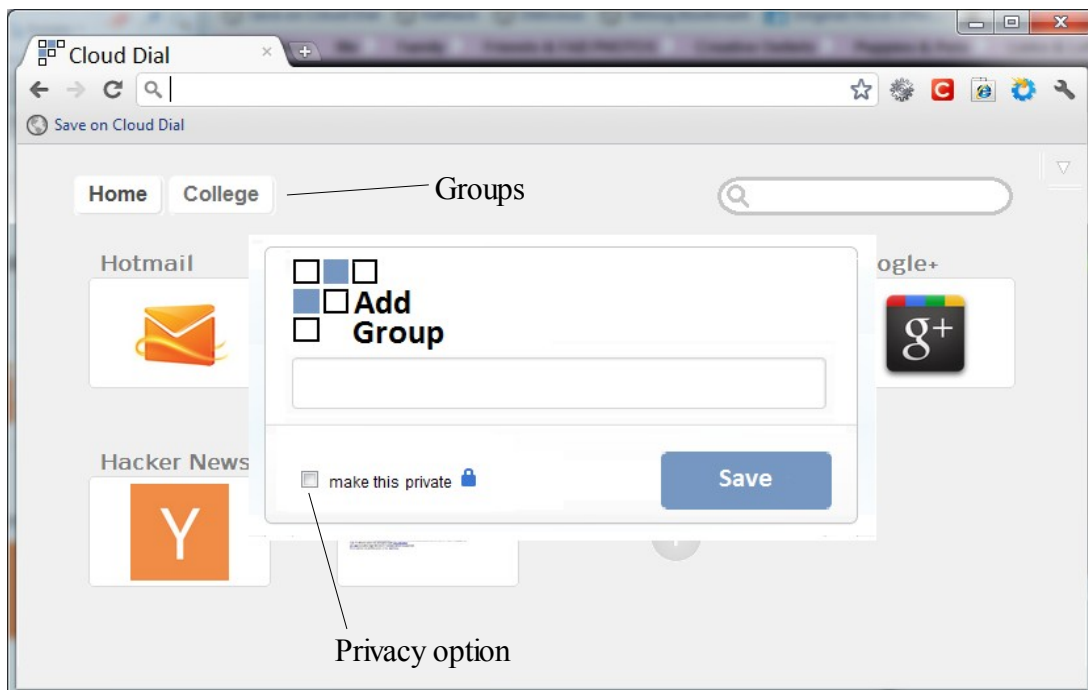
A user decides to categorise their bookmarks to improve browsing. They select the create group option on their Cloud Dial homepage and specify the name of the desired group. On addition of future bookmarks they have the option of adding to this newly created group.

2.4 Constraints

<u>Testing:</u>

There a numerous factors that will affect the testing of Cloud Dial. The system will be built with the intention of supporting thousands of users. Obviously getting this many users for testing is unrealistic, but even simulating this is extremely difficult. As such, numerous functionalities may not be tested adequately. Searching and intelligent suggestions especially depend on having a large user base.

<u>Amazon Elastic Cloud Compute:</u>

For development I will avail of Amazon's EC2 system. I will be using the free tier of this system. As such I will limited by what's offered on this tier.

<u>Programming Language Limitations:</u>

Cloud Dial will be developed almost entirely in **Javascript**. As such, it will be limited to the capabilities of this language.

<u>Time:</u>

There is much scope for the development of this project, however it limited by the time allocated to Final Year Projects by DCU.

<u>Number of Platforms:</u>

Even though the interface is being developed as a web application, having it optimised for all devices is unrealistic. This is largely due to screen sizes, with browser capabilities also being a factor. It will work best on widely used devices.

3. Functional Requirements

Listed below are the functional requirements of the proposed system. These are required to fulfil the conditions of each scenario detailed in section 2.3.

3.1 Sign up

Description:

To use Cloud Dial a user must sign up or log in through their Facebook account. If they decide instead to create a Cloud Dial specific account they must provide a username, email address and password. Upon submission of this information a confirmation email will be sent. Once confirmed their account will be activated.

Criticality:

If the user does not have a Facebook account signing up to Cloud Dial is crucial. They cannot use the system without this.

Technical Issues:

Users must trust Cloud Dial with their private information. As such it is important the sign up process is secure.

Dependencies On Other Requirements:

This is the first point of contact for a user. Therefore there are no dependencies on other requirements.

3.2 Login

Description:

A user can sign in using either their Facebook account, or a created Cloud Dial account. To do so a user submits their information in the username and password fields.

Criticality:

This is a crucial feature of Cloud Dial. Without it users cannot be differentiated, and thus cannot have their own bookmark collection.

Technical Issues:

Like the sign up process, it is also important logging in is secure as possible.

Dependencies On Other Requirements:

A user cannot login if they have not signed up for Cloud Dial, or do not have a Facebook account.

3.3 Bookmark addition

Description:

A user can add a bookmark via the addition option on their account home page, or using a Cloud Dial bookmarklet. Either option presents the user with the same addition screen, pre-populated with the website in question if using the bookmarklet.

Criticality:
Obviously this is necessary to the whole project, it is the principal functionality of the system.

Technical Issues:

Bookmarks will have a lot of information associated with them. It is important to store this information efficiently, while also ensuring it is quick to access.

Dependencies On Other Requirements:

This option will not be available to users who are not logged in.

3.4 Bookmark identification

Description:

Before saving a bookmark a user can choose an image to represent it, textual tags describing its contents, whether it's public or private, or summarise the information it contains. Cloud Dial will make suggestions for each of these. The user can also choose what group the bookmark belongs to for the purposes of categorisation.

Criticality:

A user does not have to provide this information. However doing so will aid the appearance and searching of bookmarks.

Technical Issues:

There are quite a lot of options here for users. The addition screen must have all the necessary functionality, while being intuitive to use and visually appealing. Doing so with so many choices will be challenging.

Dependencies On Other Requirements:

Providing suggestions to users depends on Cloud Dial's intelligent suggestions functionalities. However this is not necessarily needed. The user can manually select this information without it.

3.5 Tag Suggestion

Description:

When adding a bookmark Cloud Dial will suggest tags to describe its content.

Criticality:

This is not critical as a user can independently select tags. However this will aid the tagging task significantly, and hopefully increase the accuracy and amount of user tags. This will massively improve searching.

Technical Issues:

How will Cloud Dial know which tags are relevant to the bookmark in question? Some possibilities are the most prevalent terms in the document, tags used by previous users, and terms within the page's title. While all of these are reasonable solutions, each have their own issues.

Dependencies On Other Requirements:

None.

3.6 Image Suggestion

Description:

When adding a bookmark Cloud Dial will suggest images to represent it.

Criticality:

This is not crucial. However it may aid bookmark browsing, on mobile devices in particular. It will also make a users account more visually appealing.

Technical Issues:

How will Cloud Dial know what image best represents a particular bookmark? A reasonable solution would be a web page screen shot, however these are not always useful for identifying particular pages. Again the possibility of using those chosen by other users is there, however this will not be possible in the case where the bookmark in question has not been previously added.

Another possibility is to use an existing image search tool, using the bookmark's domain as the search criteria. This would work quite well, but again falls down when a user has multiple bookmarks from the same domain. One solution would be to superimpose the bookmark's tags on the suggested image in order to individualise it. How well this works though, is yet to be seen.

Dependencies On Other Requirements:

If it's decided to use the tags in an attempt to individualise images, it would be helpful if the tag suggestion functionality is working well.

3.7 Summary Suggestion

<u>Description:</u>

When adding a bookmark Cloud Dial will summarise the bookmark's contents. The user can change this summary if they so choose.

<u>Criticality:</u>

This is not crucial. However it may improve bookmark searching.

<u>Technical Issues:</u>

Summarising content is a long existing challenge. How best to do this will be tough choice.

<u>Dependencies On Other Requirements:</u>

None.

3.8 Bookmark search

<u>Description:</u>

A user will provide a plaintext query when searching for a bookmark. If they wish to refine the search Cloud Dial will suggest additional query terms. Multiple ordering options will also be offered.

<u>Criticality:</u>

For users who amass a large amount of bookmarks, searching is crucial. Without it finding a desired bookmark would be tedious and time consuming.

<u>Technical Issues:</u>

There are numerous options for effective searching, selecting the one which works best will undoubtedly be challenging.

<u>Dependencies On Other Requirements:</u>

For Cloud Dial's searching functionality to work well intelligent suggestions should be available to users. Otherwise a user may not detail their bookmarks accurately.

3.9 Facebook sharing

Description:

If a user signs in using their Facebook account they have an option of automatically sharing their bookmark adding activities. If a user chooses this option their Facebook account will be updated with the details of each bookmark automatically upon addition.

Criticality:

This is not at all crucial, but user sharing will likely increase Cloud Dial's popularity.

Technical Issues:

This development relies on Facebook's Open Graph API. It will be limited by the capabilities of this.

Dependencies On Other Requirements:

This depends on the user logging in through Facebook.

3.10 Categorisation

Description:

Users have the option of categorising their bookmarks. To do this they must create groups or sub-groups. When created a user has the option of saving to these locations during bookmark addition.

Criticality:

For users with a large number of bookmarks this is critical. Without it the user experience will be badly effected, and may result in lost custom.

Technical Issues:

For desktop applications such functionality is trivial to develop. However developing this within a browser with the optimal user experience in mind may be challenging.

Dependencies On Other Requirements:

None.

4. System Architecture

4.1 System Architecture Diagram
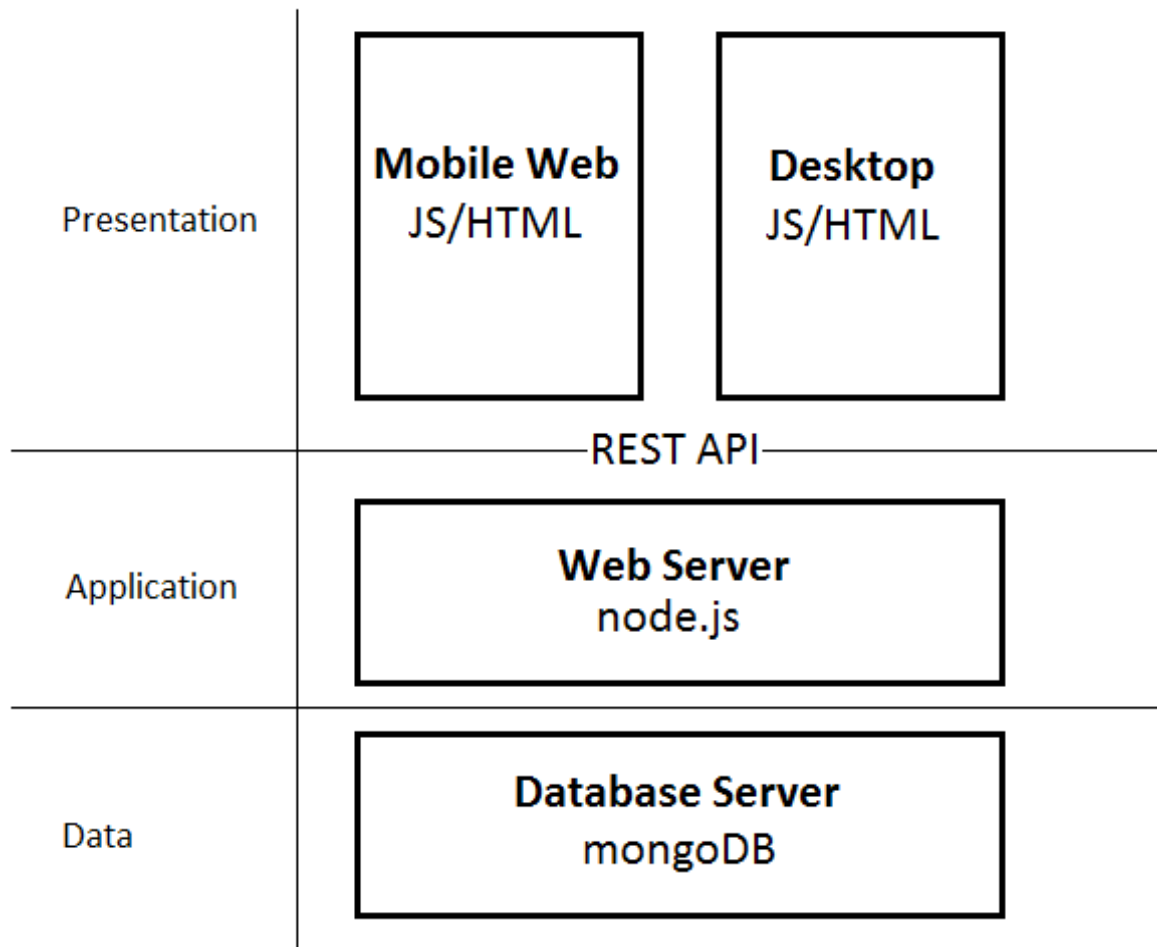
Fig 4.1

Fig 4.1 Is a high level representation of the architecture of Cloud Dial. It shows clearly the 3 categories it can be broken into.

4.2. System Architecture Description

4.2.1 Presentation tier

The presentation layer is concerned only with presentation. It has two different components. One for desktop devices, and another for mobile and touch devices. These are treated separately as the presentation for each is very different.

Both will be developed using **HTML5**, the **MVC** framework **backbone.js,** and **JQuery** for the interface logic.

4.2.2 Application tier

The application layer contains the logic of the application. It controls the application's functionality, and is central to its operation.

It contains the web server which will be developed using **node.js**. Node will be used primarily for client server communications, but may also be used for the business logic of Cloud Dial, unless performance suffers as a result.

4.2.3 Data tier

The data layer contains the application's database. This is where all user information is stored and retrieved.

This will be developed using **mongodb** primarily for its scalability, which is extremely important for cloud based systems.

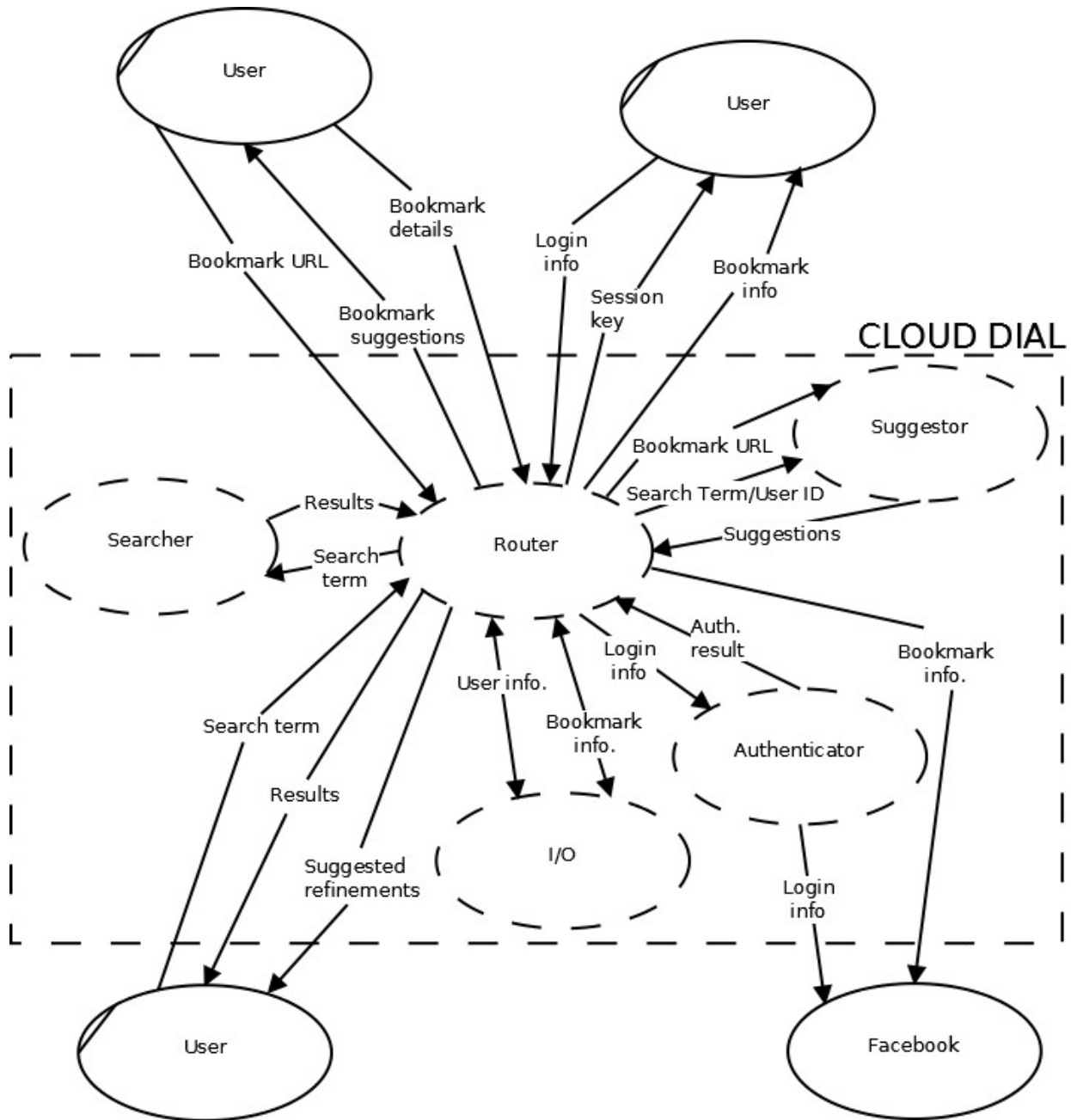# 5. High-Level Design

5.1 High-Level Design Diagram

Fig 5.1

Fig 5.1 is a high level data flow diagram(DFD) of Cloud Dial. It shows its main components, both internally and externally.

5.2 High Level Design Description

5.2.1 User

This represents an active user of Cloud Dial. The DFD considers several different user types. These are external to Cloud Dial.

5.2.2 Facebook

Facebook can be used for authentication. For this Cloud Dial must communicate user provided login information with Facebook.

Users can also allow automatic updates to their Facebook account on addition of bookmarks. For this bookmark information must be provided.

This is external to Cloud Dial.

5.2.3 Router

This is central to all functionality in Cloud Dial. Requests are sent, and are routed by the Router to the appropriate component. This is internal to Cloud Dial.

5.2.4 Authenticator

The Authenticator is used by Cloud Dial to authenticate users on login. It uses either Facebook or Cloud Dial's own records to do this, and if approved a session key is passed. This is internal to Cloud Dial.

5.2.5 Suggester

The Suggester is responsible for making intelligent suggestions to a user. Namely tag suggestions, image suggestions, summary suggestions, and search refinement suggestions. To do this the component needs the URL of a bookmark being added. Using this it can analyse the content and make suggestions based on underlying assumptions. This is an internal component.

5.2.6 Searcher

The searcher component is used for bookmark searching. To do this the component needs a query and a search space. This information will be routed to the Searcher component on the event of a search. This is internal to the system.

5.2.7 I/O

The I/O component is responsible for storing and retrieving user information. It will interface with Cloud Dial's database, and communicate with the Router. It is internal to Cloud Dial.

6. Preliminary Schedule

Name	Begi...	End date
Submit functional spec	12/5/11	12/6/11
Write functional spec	11/7/11	12/6/11
Basic front end dev	11/21...	12/10/11
Database dev	12/12...	12/17/11
Basic server dev	12/19...	12/24/11
Front end/server integration	1/2/12	1/14/12
Exams	1/16/12	1/28/12
Finish incomplete work	1/30/12	2/11/12
Intelligent suggestions dev	2/13/12	2/25/12
Front end++/Touch screen support	2/27/12	3/17/12
Facebook open graph	3/19/12	3/24/12
Search improvements	3/26/12	3/31/12
Finishing touches	4/2/12	4/28/12
Testing	4/16/12	4/28/12
Documentation	4/23/12	5/5/12
Submit project documentation	5/11/12	5/12/12
Exams	5/14/12	5/26/12
Interface improvements	5/28/12	6/1/12
Project expo	5/29/12	5/30/12
Project demo	5/30/12	6/1/12

7. References

- ➢ **Online Advertising Expenditure Forecast, 2009-2010**

  [Internet] Clickz

  Available from: <http://www.clickz.com/clickz/stats/1708807/online-advertising-expenditure-forecast-2009-2010>

  [Last accessed: 04/12/11]

- ➢ [Internet] Foldoc

  Available from: <http://foldoc.org/>

  [Last accessed: 04/12/11]

- ➢ [Internet] Amazon

  Available from: <http://aws.amazon.com/ec2/>

  [Last accessed: 04/12/11]

- ➢ [Internet] W3C

  Available from: <http://www.w3.org/>

  [Last accessed: 4/12/11]

- ➢ [Internet] Backbone.js

  Available from: <http://documentcloud.github.com/backbone/>

  [Last accessed: 4/12/11]

- ➢ [Internet] jQuery

  Available from: <http://jquery.com/>

  [Last accessed: 4/12/11]

- ➢ [Internet] node.js

  <http://nodejs.org/>

  [Last accessed: 4/12/11]

- ➢ [Internet] MongoDB

  Available from: <http://www.mongodb.org/>

  [Last accessed: 4/12/11]