MY TRIBE-A NETWORK OF PROFESSIONALS

Anindita Bhowmik, Veeramani R.

Undergraduate Student, Information Technology, SRM University, Tamil Nadu, India

Assistant Professor, Information Technology, SRM University, Tamil Nadu, India

ABSTRACT

The beginning of a tribe of professionals offering personalized services or products. It does not end at one point, in fact it cannot end at any point - owing to the increase in number of services needed and professionals to cater to the needs of everyday life and the world.

The world's a stage and how about you get your things done just by a click here, from the comfort of your home?

Check out profiles under different sections, have conversations with them, talk to them about your needs and book them for a minimal charge. You go there!

We also provide some special projects like handloom products and medical products trading to countries which needs them through partnerships, getting retired professionals to connect to the youth and mentor them, thus spreading their bit of knowledge with the world and never stopping anywhere.

All of these can be found in the same platform, My Tribe website and an android application.

1. INTRODUCTION

Our personal experiences made us realize that even in 2017, hiring local service providers is actually extremely hard and difficult—whether it is a reliable plumber or a reliable yoga trainer or a reliable wedding photographer. To search for an electrician or a plumber, people mostly ask friends and neighbours or flip through yellow pages.

Thus, a vision of going beyond a mere search and discovery platform by building a business model that encompasses on-boarding service providers; an on-demand service ecosystem to hire local services at the push of a button, from the comfort of home. The idea is to redefine how local services and serviceman are being hired in India, after seeing opportunity in the broken system of how the country is connected with various service providers.

1.1. Purpose of the Project

MyTribe is about modifying education through integration of technology and social connect, in the society. We are getting down to the needs of students and employees in the network of Ramapuram, especially people who have just migrated, thus reaching to the grassroots level.

2. SYSTEM ANALYSIS

2.1 Problem in Existing System

- Problem with current scenario

Many of the people we work with are executives and other professionals who are very successful in their careers but between their job and their family, they often have little time left over for hiring s[i]ervicemen for different day-to-day activities, especially when many services seem as urgent as other things going on in their lives. In that case, managing end-to-end personal services can feel like fighting one-handed.

By the same token, we all have handicaps when it comes to managing our conditions or how we stay. Fortunately, they are a bit easier to overcome than having to strike someone with our elbow. Here are three of the most common limitations and ways to overcome them:

- "I can't afford to waste time on getting things fixed or managing them."
- "I'm wondering if he the mechanic is over-charging me or that is the usual rate."
- "I don't have the time or knowledge to call up an electrician for fixing my AC. I have just shifted to Lucknow and I work most of the times in my office."

- Solutions to these problems

I have always wanted to solve a really big consumer problem in India by leveraging mobile technology. MyTribe's matchmaking engine sets up potential customers with the best service providers in their area, saving them from the tedium of going through hundreds of listings or having to rely on word-of-mouth referrals.

2.1 Proposed System

To use MyTribe, a customer first selects a category and then answers several questions on its Android app or website. Then the platform's recommendation engine uses that information and location-based data to match him or her with service providers.

For example, if someone wants to study guitar, the questions include what genre of music they are interested in, budget, schedule, and skill level. Then the request is matched to a handful of teachers, who respond in-app if they are interested with price quotes.

Professionals have to apply to be listed and are rated on a five-star scale by customers, which helps maintain quality. MyTribe uses the same process and technology to deliver results for categories ranging from salon services to home cleaning.

2.3 Modules and their description

This application comprises of 4 Modules:

1. Register

2. Login

3. Create profile

4. Enter the details

2.3.1Description:

- Register: User first need to register into the system simply by filling up the details such as Name, Email id & Phone number.
- Set password: After filling up the details, user can now set a password of his/her choice for security purpose.

- Login: After successful registration, user can now login into the system by entering the password.
- Check professionals (for users): Users can now search for services and professionals and contact them.
- Apply for a job (for professionals): Professionals can apply for jobs and be a partner on acceptance.

3. PRODUCT DESIGN

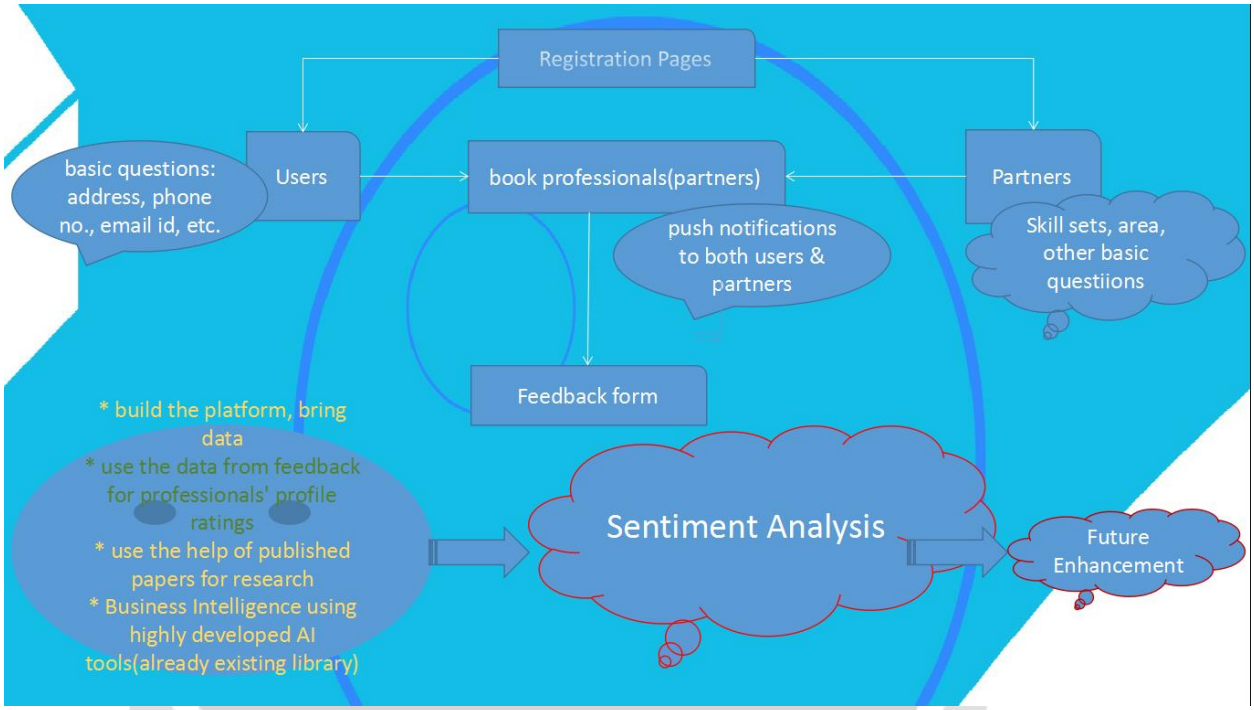


Fig-1: System Architecture

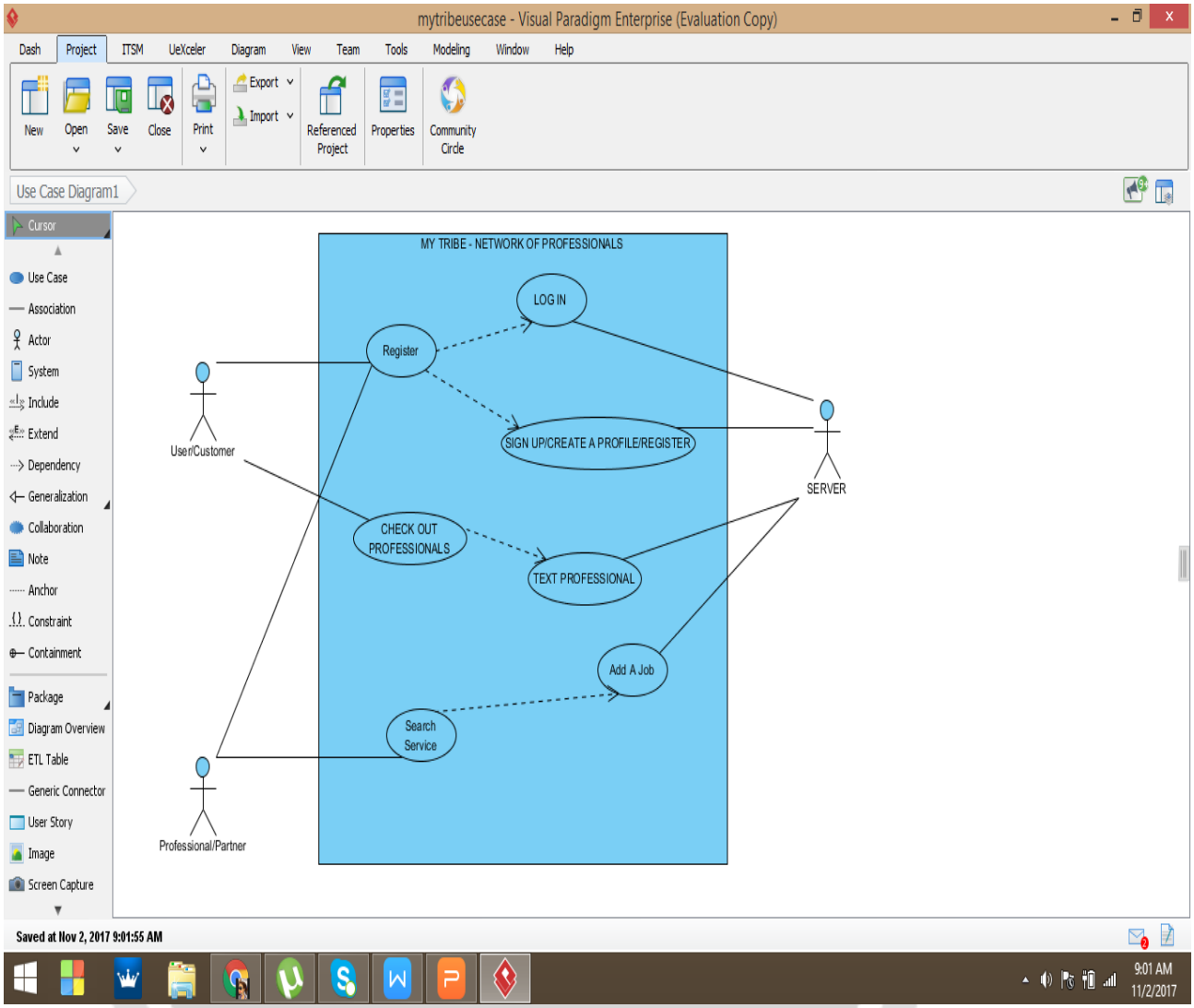


Fig-2: Use-case Diagram

4. PROJECT IMPLEMENTATION

4.1 Introduction

Eclipse IDE has been used to develop this project. We used website development for Design and coding of project. Maintained all databases into MySQL Server, in that we create tables, write query for store data or record of project. The application is database-driven, and shall be deployed on Google Cloud platform.

4.2 Software Requirement:

- Windows XP, Windows 7(ultimate & enterprise) or any
- PHP 7, Java 9
- Eclipse Oxygen
- Spring 5
- MyBatis
- MySQL 5.7

- Apache Tomcat 9
- Android Studio

4.3 Overview of Technologies used

4.3.1 Front End technology

4.3.1.1 Introduction to Android

Android provides a rich application framework that allows you to build innovative apps and games for mobile devices in a Java language environment. The documents listed in the left navigation provide details about how to build apps using Android's various APIs.

If you're new to Android development, it's important that you understand the following fundamental concepts about the Android app framework:

4.3.1.2 Apps provide multiple entry points

Android apps are built as a combination of distinct components that can be invoked individually. For instance, an individual *activity* provides a single screen for a user interface, and a *service* independently performs work in the background.

From one component you can start another component using an *intent*. You can even start a component in a different app, such as an activity in a maps app to show an address. This model provides multiple entry points for a single app and allows any app to behave as a user's "default" for an action that other apps may invoke.

4.3.1.3 Apps adapt to different devices

Android provides an adaptive app framework that allows you to provide unique resources for different device configurations.

For example, you can create different XML layout files for different screen sizes and the system determines which layout to apply based on the current device's screen size. You can query the availability of device features at runtime if any app features require specific hardware such as a camera. If necessary, you can also declare features your app requires so app markets such as Google Play Store do not allow installation on devices that do not support that feature.

4.3.1.4 Application Fundamentals

Android apps are written in the Java programming language. The Android SDK tools compile your code along with any data and resource files into an APK: an Android package, which is an archive file with an .apk suffix. One APK file contains all the contents of an Android app and is the file that Android-powered devices use to install the app.

Once installed on a device, each Android app lives in its own security sandbox:

- The Android operating system is a multi-user Linux system in which each app is a different user.
- By default, the system assigns each app a unique Linux user ID (the ID is used only by the system and is unknown to the app). The system sets permissions for all the files in an app so that only the user ID assigned to that app can access them.
- Each process has its own virtual machine (VM), so an app's code runs in isolation from other apps.
- By default, every app runs in its own Linux process. Android starts the process when any of the app's components need to be executed, then shuts down the process when it's no longer needed or when the system must recover memory for other apps.

In this way, the Android system implements the *principle of least privilege*. That is, each app, by default, has access only to the components that it requires to do its work and no more. This creates a very secure environment in which an app cannot access parts of the system for which it is not given permission.

4.3.1.5 Declaring components

The primary task of the manifest is to inform the system about the app's components. For example, a manifest file can declare an activity as follows:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest ... >
    <application android:icon="@drawable/app_icon.png" ... >
        <activity android:name="com.example.project.ExampleActivity"
                  android:label="@string/example_label" ... >
        </activity>
        ...
    </application>
</manifest>
```

In the <application> element, the android:icon attribute points to resources for an icon that identifies the app.

In the <activity> element, the android:name attribute specifies the fully qualified class name of the Activity subclass and the android:label attributes specifies a string to use as the user-visible label for the activity.

You must declare all app components this way:

- <activity> elements for activities
- <service> elements for services
- <receiver> elements for broadcast receivers
- <provider> elements for content providers

Activities, services, and content providers that you include in your source but do not declare in the manifest are not visible to the system and, consequently, can never run. However, broadcast receivers can be either declared in the manifest or created dynamically in code (as BroadcastReceiver objects) and registered with the system by calling registerReceiver()

4.3.1.6 Declaring component capabilities

As discussed above, in Activating Components, you can use an Intent to start activities, services, and broadcast receivers. You can do so by explicitly naming the target component (using the component class name) in the intent. However, the real power of intents lies in the concept of *implicit intents*. An implicit intent simply describes the type of action to perform (and, optionally, the data upon which you'd like to perform the action) and allows the system to find a component on the device that can perform the action and start it. If there are multiple components that can perform the action described by the intent, then the user selects which one to use.

The way the system identifies the components that can respond to an intent is by comparing the intent received to the intent filters provided in the manifest file of other apps on the device.

When you declare an activity in your app's manifest, you can optionally include intent filters that declare the capabilities of the activity so it can respond to intents from other apps. You can declare an intent filter for your component by adding an <intent-filter> element as a child of the component's

4.3.1.7 What is SEO? Search Engine Optimization for hyper-local service pages

SEO is all the things you have to do to get your website ranked high in search results, without paying for it. SEO is about optimizing a web page, with the end goal of ranking high (page one) on a search engine like Google.

Google's number one goal is to show its users most relevant content. A company's goal is usually to rank on top in search results. If the company helps meet Google's goals, Google will help meet the company's goals!

Fig-3: SEO characteristics

Yes, SEO could be simple to crack if one really understands what Google needs.

4.3.1.7.1 MyTribe and SEO

SEO is a major source of organic traffic at MyTribe therefore its important to optimize SEO pages for Google Crawler Engine.

MyTribe's content is unique in a lot of ways. Its itinerary consists of people themselves. MyTribe showcases from an Engineer who is a freelancing photographer to plumbers to fitness trainers.

MyTribe's SEO pages are spread across partner's buckets and hyper localities.

Typically a bucket's page is divided into:

- Hyper-local pages
- City pages

MyTribe strives to achieve the following for its SEO pages :

- Speed

- Mobile friendly
- High quality
- Rich content

SEO hyper-local/city page architecture

MyTribe SEO pages are divided into:

- Product (here professionals/partner) listing
- Meta data
- Historic data
- Quick-links

Product listing: Page's partner listing is generated via mutable algorithms. Every page is tagged to its hyper-local details. Using this information, algorithm gather all relevant partners for and around a location. Ever increasing data set poses a few challenges and hence the algorithm keeps going through new enhancements.

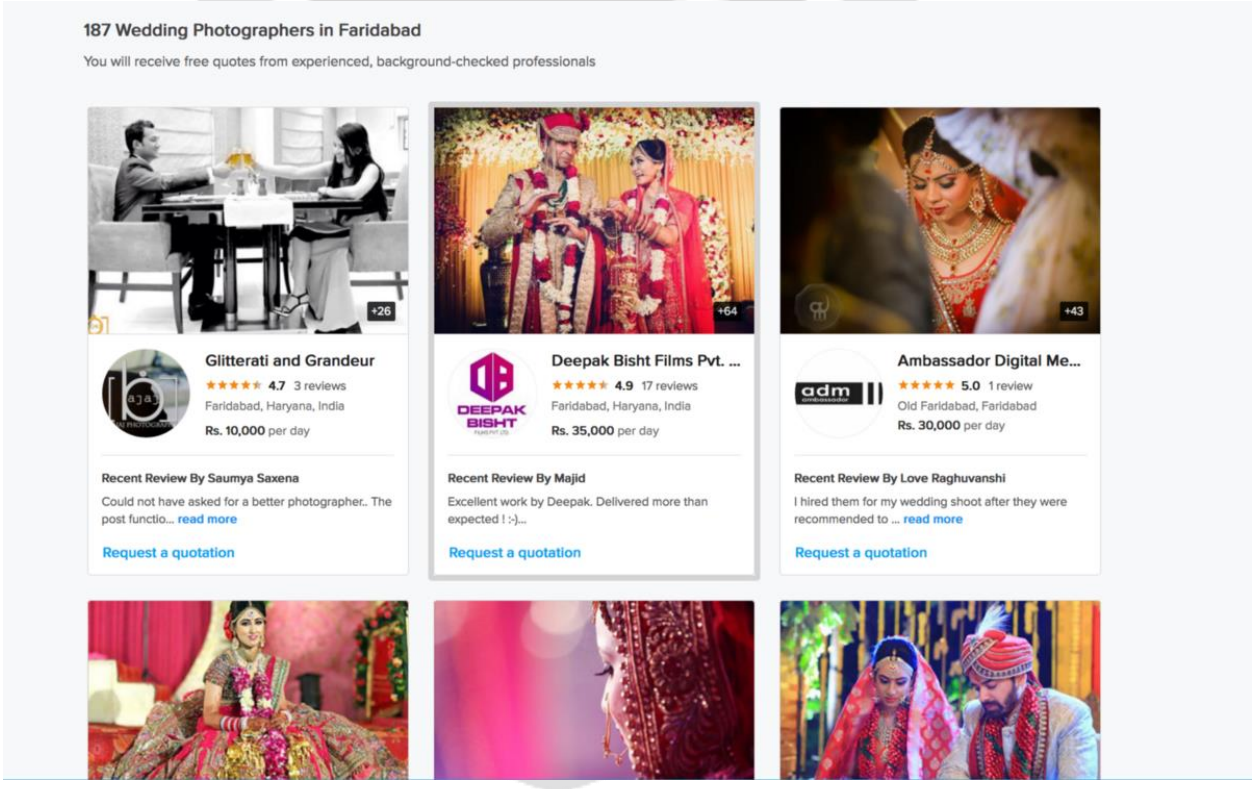
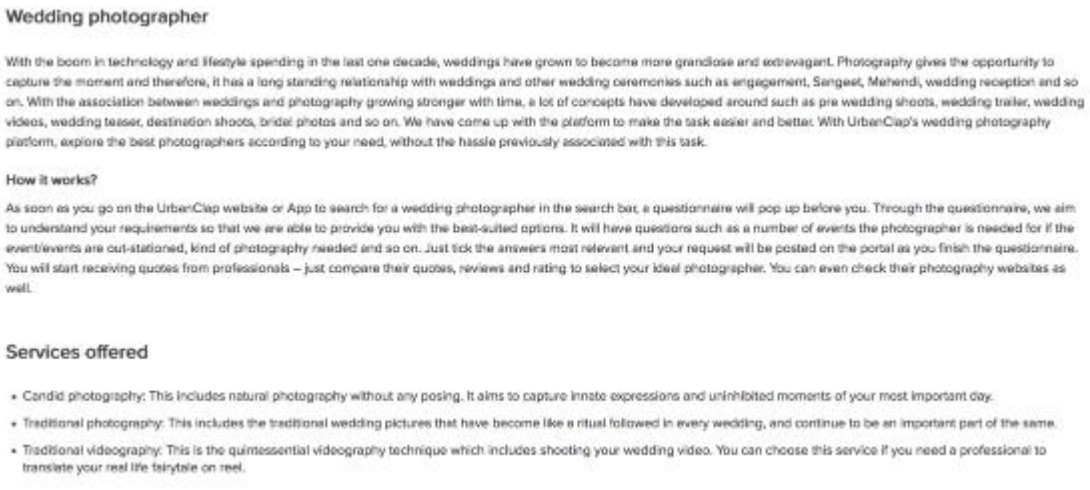


Fig-4: Product Listing

Meta-Data: Page's meta data is static or dynamic information, related to a location and bucket. It is essential to meet Google standards in creating this data. Meta data not only provides relevant information to the customers, but also holds ranking keywords for Google's crawl engine. Meta data gets richer through weekly analysis of webmaster tool's reports.



Wedding photographer

With the boom in technology and lifestyle spending in the last one decade, weddings have grown to become more grandiose and extravagant. Photography gives the opportunity to capture the moment and therefore, it has a long standing relationship with weddings and other wedding ceremonies such as engagement, Sangeet, Mehendi, wedding reception and so on. With the association between weddings and photography growing stronger with time, a lot of concepts have developed around such as pre wedding shoots, wedding trailer, wedding videos, wedding teaser, destination shoots, bridal photos and so on. We have come up with the platform to make the task easier and better. With UrbanClap's wedding photography platform, explore the best photographers according to your need, without the hassle previously associated with this task.

How it works?

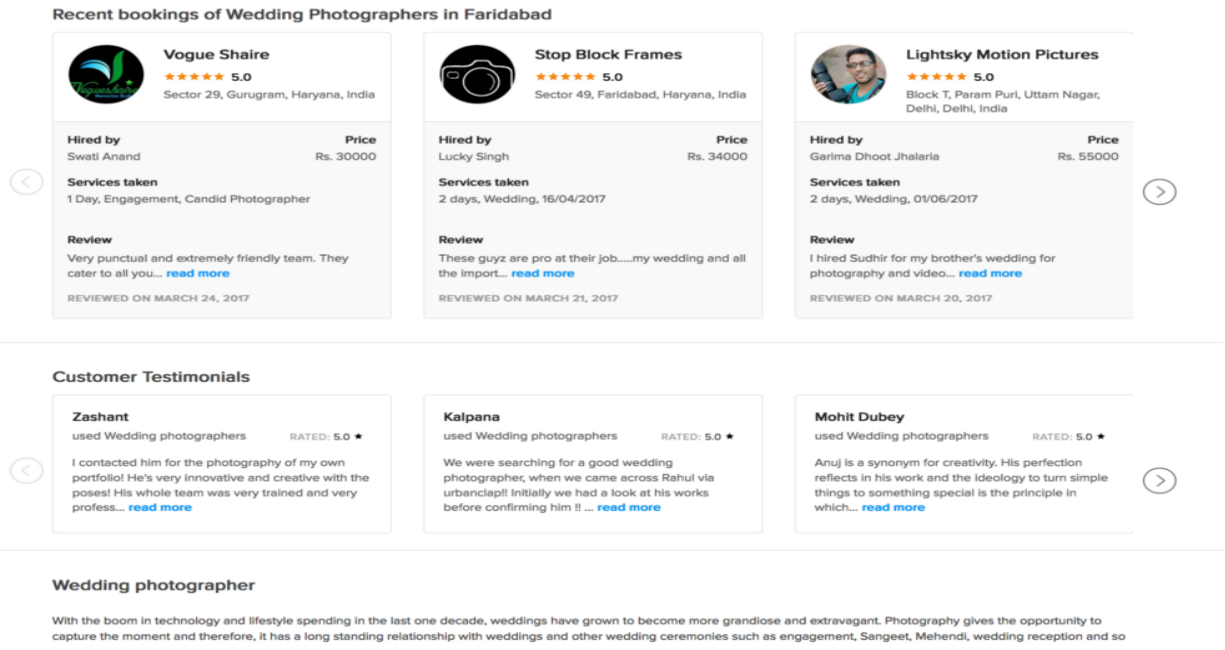
As soon as you go on the UrbanClap website or App to search for a wedding photographer in the search bar, a questionnaire will pop up before you. Through the questionnaire, we aim to understand your requirements so that we are able to provide you with the best-suited options. It will have questions such as a number of events the photographer is needed for if the event/events are out-stationed, kind of photography needed and so on. Just tick the answers most relevant and your request will be posted on the portal as you finish the questionnaire. You will start receiving quotes from professionals – just compare their quotes, reviews and rating to select your ideal photographer. You can even check their photography websites as well.

Services offered

- Candid photography: This includes natural photography without any posing. It aims to capture innate expressions and uninhibited moments of your most important day.
- Traditional photography: This includes the traditional wedding pictures that have become like a ritual followed in every wedding, and continue to be an important part of the same.
- Traditional videography: This is the quintessential videography technique which includes shooting your wedding video. You can choose this service if you need a professional to translate your real life fairytale on reel.

Fig-5: Meta-data

Historic Data: Past interactions of customers with the platform in the specific location bucket is collected, curated and presented on the SEO pages. Any such data will be unique to platform and is a good source of keywords too.



Recent bookings of Wedding Photographers in Faridabad

Vogue Shaire
★★★★★ 5.0
Sector 29, Gurugram, Haryana, India

Hired by: Swati Anand
Price: Rs. 30000
Services taken: 1 Day, Engagement, Candid Photographer
Review: Very punctual and extremely friendly team. They cater to all you... read more
REVIEWED ON MARCH 24, 2017

Stop Block Frames
★★★★★ 5.0
Sector 49, Faridabad, Haryana, India

Hired by: Lucky Singh
Price: Rs. 34000
Services taken: 2 days, Wedding, 16/04/2017
Review: These guyz are pro at their job.....my wedding and all the import... read more
REVIEWED ON MARCH 21, 2017

Lightsky Motion Pictures
★★★★★ 5.0
Block T, Param Puri, Uttam Nagar, Delhi, Delhi, India

Hired by: Garima Dhoot Jhalaria
Price: Rs. 55000
Services taken: 2 days, Wedding, 01/06/2017
Review: I hired Sudhir for my brother's wedding for photography and video... read more
REVIEWED ON MARCH 20, 2017

Customer Testimonials

Zashant
used Wedding photographers RATED: 5.0 ★
I contacted him for the photography of my own portfolio! He's very innovative and creative with the poses! His whole team was very trained and very profess... read more

Kalpana
used Wedding photographers RATED: 5.0 ★
We were searching for a good wedding photographer, when we came across Rahul via urbanclap!! Initially we had a look at his works before confirming him !! ... read more

Mohit Dubey
used Wedding photographers RATED: 5.0 ★
Anuj is a synonym for creativity. His perfection reflects in his work and the ideology to turn simple things to something special is the principle in which... read more

Wedding photographer

With the boom in technology and lifestyle spending in the last one decade, weddings have grown to become more grandiose and extravagant. Photography gives the opportunity to capture the moment and therefore, it has a long standing relationship with weddings and other wedding ceremonies such as engagement, Sangeet, Mehendi, wedding reception and so

Fig-6: Historic Data

Quick Links: For a crawler engine to efficiently crawl and rank pages, the website should have a spider web like structure. Such structures are possible through smart and intensive interlinking. Linking to nearby hyper-local pages across same or different buckets of categories helps in an effective user navigation, as well as page interlinking and indexing.

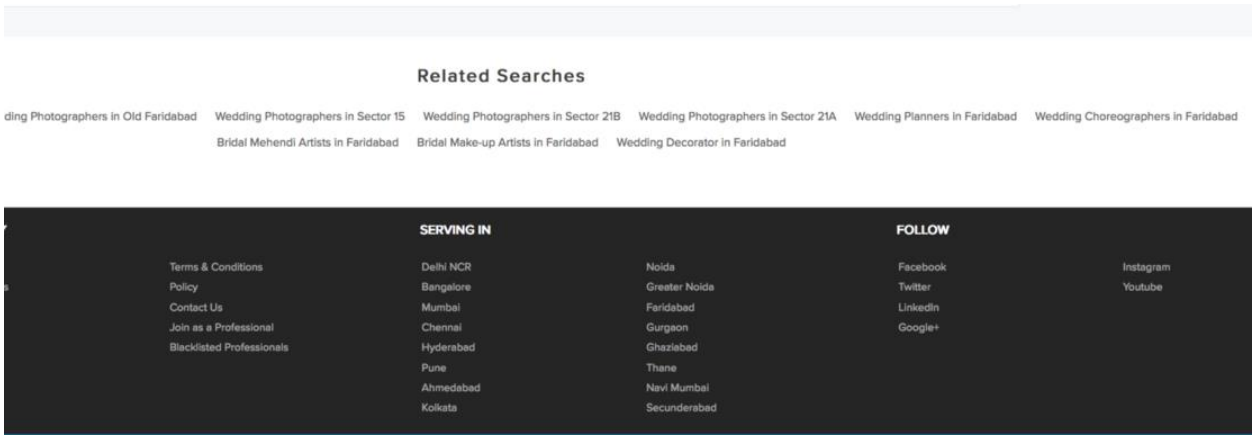


Fig-7: Quick Links

4.3.1.7.2 SEO caching—speed vs content

Google assigns variable interval of crawl time to every web domain, say t. One of the primary aim is to make Google crawl maximum pages in t. More pages Google can crawl in t, better it ranks the pages (assuming good quality, relevant data and proper interlinking).

For most simple get requests, this will not be a challenge. However, when complex algos run on huge data sets, speed takes an adverse impact.

To optimize for speed, any API that does heavy processing, is pre-processed and cached , ready to serve. This consequentially introduces another issue, dynamics of the content.

Good ranking SEO pages have time dependent content. Example: Updating partners on the pages as new ones are added to the platform.

To solve for this, all data entry points are hooked to the services that pre-processes and caches responses. Example: if a new partner is added to salon at home for women in Delhi, all hyper-local pages near that partner will incorporate this change.

Any change in data is hooked to exposed services, which refresh the cache as well.

This solves for optimal speed and latest relevant content.

4.3.1.7.3 Keeping up with business (Fault tolerance)

Google marks down pages which return 404 (not found) errors. MyTribe is in itself an ever changing ecosystem, with new locality or category buckets being added or disabled from the platform.

The challenge imposed in such ecosystem is to elegantly inform Google of such bucket being disabled or making sure Google does not crawl these pages.

Generating new sitemaps on daily basis and plugging broken web links is one way to do it. However disabled bucket's pre-indexed pages will always show up in searches. For such scenarios, soft 404 is returned. Soft 404 is a fancy way to show a 404 page, with alternate options for the customer to navigate too. Google doesn't penalize soft 404. Redirection (302) is also done for such pages and is acceptable by Google as well to a point.

Disabled buckets are handled with sitemap, soft 404 or 302 redirects

This solves for ever changing business.

4.3.1.7.4 Dynamic page generation (Scalability)

As discussed above, new buckets of category-locality-city are added every day to MyTribe. Example: MyTribe is recently going live in Kolkata. To cope up with high business demand, challenge is to generate SEO relevant data at thought.

New entries are created in the database. Page contents are simultaneously pre-processed and cached.

MyTribe's SEO fresh page generation logic has been abstracted to a single click, at the new bucket's internal UI.

4.3.1.7.5 To the future

MyTribe is working on enhancements and additions for its SEO pages. Some of them are listed below:

- Front End—Mobile and Desktop
- Separate server units for SEO
- Scaling cache
- HQ Media

SEO team continuously hustle for better content, richer quality, fast rendering and good speed. Battle for SEO services' pages is being fought everyday from the MyTribe Desks!

4.3.2 BACK END TECHNOLOGY:

4.3.2.1 About MySQL Server

MySQL is the world's most popular open source database. With its proven performance, reliability, and ease-of-use, MySQL has become the leading database choice for web-based applications, used by high profile web properties including Facebook, Twitter, YouTube. Additionally, it is an extremely popular choice as embedded database, distributed by thousands of ISVs and OEMs.

Built on MySQL Enterprise Edition and powered by Oracle Cloud, Oracle MySQL Cloud Service provides a simple, automated, integrated, and enterprise-ready cloud service enabling organizations to increase business agility and reduce costs.

Database:

A database is similar to a data file in that it is a storage place for data. Like a data file, a database does not present information directly to a user; the user runs an application that accesses data from the database and presents it to the user in an understandable format.

A database typically has two components: the files holding the physical database and the database management system (DBMS) software that applications use to access data. The DBMS is responsible for enforcing the database structure, including:

- Maintaining the relationships between data in the database.
- Ensuring that data is stored correctly and that the rules defining data relationships are not violated.
- Recovering all data to a point of known consistency in case of system failures.

4.3.2.2 Relational Database

There are different ways to organize data in a database but relational databases are one of the most effective. Relational database systems are an application of mathematical set theory to the problem of effectively organizing data. In a relational database, data is collected into tables (called relations in relational theory).

When organizing data into tables, you can usually find many different ways to define tables. Relational database theory defines a process, normalization, which ensures that the set of tables you define will organize your data effectively.

4.3.2.3 Client/Server

In a client/server system, the server is a relatively large computer in a central location that manages a resource used by many people. When individuals need to use the resource, they connect over the network from their computers, or clients, to the server.

Examples of servers are: In a client/server database architecture, the database files and DBMS software reside on a server. A communications component is provided so applications can run on separate clients and communicate to the database server over a network. The SQL Server communication component also allows communication between an application running on the server and SQL Server.

Server applications are usually capable of working with several clients at the same time. SQL Server can work with thousands of client applications simultaneously. The server has features to prevent the logical problems that occur if a user tries to read or modify data currently being used by others.

While SQL Server is designed to work as a server in a client/server network, it is also capable of working as a stand-alone database directly on the client. The scalability and ease-of-use features of SQL Server allow it to work efficiently on a client without consuming too many resources.

4.3.2.4 Structured Query Language (SQL)

To work with data in a database, you must use a set of commands and statements (language) defined by the DBMS software. There are several different languages that can be used with relational databases; the most common is SQL.

Both the American National Standards Institute (ANSI) and the International Standards Organization (ISO) have defined standards for SQL. Most modern DBMS products support the Entry Level of SQL-92, the latest SQL standard (published in 1992).

4.3.2.5 SQL Server Features

Microsoft SQL Server supports a set of features that result in the following benefits:

i) Ease of installation, deployment, and use

SQL Server includes a set of administrative and development tools that improve your ability to install, deploy, manage, and use SQL Server across several sites.

ii) Scalability

The same database engine can be used across platforms ranging from laptop computers running Microsoft Windows® 95/98 to large, multiprocessor servers running Microsoft Windows NT®, Enterprise Edition.

iii) Data warehousing

SQL Server includes tools for extracting and analyzing summary data for online analytical processing (OLAP). SQL Server also includes tools for visually designing databases and analyzing data using English-based questions.

iv) System integration with other server software

SQL Server integrates with e-mail, the Internet, and Windows.

4.3.2.6 Databases

A database in Microsoft SQL Server consists of a collection of tables that contain data, and other objects, such as views, indexes, stored procedures, and triggers, defined to support activities performed with the data.

The data stored in a database is usually related to a particular subject or process, such as inventory information for a manufacturing warehouse.

SQL Server can support many databases, and each database can store either interrelated data or data unrelated to that in the other databases. For example, a server can have one database that stores personnel data and another that stores product-related data. Alternatively, one database can store current customer order data, and another; related database can store historical customer orders that are used for yearly reporting. Before you create a database, it is important to understand the parts of a database and how to design these parts to ensure that the database performs well after it is implement.

4.3.2.7 MVC, ReST, SOA, functional programming – all together!

4.3.2.7.1 Introduction

As software engineers, we work in teams and are most often working on code that is drafted by another team mate. We end up taking days to understand the code, the logic flow, the dependencies, the constraints, the corner-cases – only to make a few hours worth of changes. The difference between teams with good and quick iteration cycles vs lethargic code bases is the quality of code written.

Startups go through various phases of scale, and hence the segmentation of legacy code are of varying quality and style. Its important to constantly re-look at what is written. There is another post which talks about the code itself, here I will simply cover the architectural journey and how / where these buzzwords come in.

4.3.2.7.2 MVC, REST, SOA – all together!

Lets take an example of a simple application—retrieve data from a database and show it on a web page.

This requires a data base (store), some code to read and manipulate the data to prepare it for presentation (controller) and the web-page code itself to display that data (view). This is quite simply the Model-View-Controller way of architecting. The model has a controller layer which serves data to the visual view layer.

Now, as this system grows, lets say you add an Android App as a client. You also want to edit/update along with get. This now requires you to explicitly decouple your server code from the 'client' code. You will need an API to interact with the data in a state-less, client-server protocol, probably over HTTP. The model is still exposed via a controller, but the controller is evolved into being a ReST-ful API exposed over HTTP for server-to-client.

Fast forward, you have multiple database models, views are getting complex with multiple data models getting pulled. This is the right time to relook at how the system has been designed. Instead of exposing models via controllers, you start to encapsulate the 'purpose' as a Service. A logical service here retrieves the data to provide useful service regardless of the underlying data store. This service exposes the data to other systems, where a client can be a system itself. The service itself is now exposed using ReST-ful architecture over HTTP for system-to-system communication. The 'model' of different data units becomes redundant / useless / incidental at the service level. This evolution is into that of a Service-Oriented-Architecture.

The client now, has been exposing the data directly. To better architecture this data-to-view architecture, we would start modelling the incoming data, maybe even storing or caching it. This is where MVC gets used on the client itself. The client stores things as models, using a controller to be the middle manipulation layer to push / pull data from the view. Over time, even this becomes complex enough to go through the above journey, where client starts having a service instead of directly dealing with models, and using controller-for-services to talk and display data to the visual layer.

Now, before we go deeper with functional programming, lets understand some context behind it.

4.3.2.7.3 What are Programming Paradigms?

Programming paradigms are a –

- "way" of programming and structuring code
- "how" you write code
- they are NOT mutually exclusive to each other
- languages DONT have paradigms
- languages might / might-not make it easy to write in some paradigms

Each paradigm gives a flavour that optimises for –

- style and syntax
- code organisation and grouping

- execution model (side effects, parallelism, etc)
- sequence of operations and triggers

Popular Programming Paradigms

- Imperative—"how" the computation takes place step-by-step, explicit control flow, go-to's, side-effects caused by global state manipulation
- Imperative > Structured—introduces more logical program structure, loops, conditions, indentation (eg. using FOR loop instead of GOTOs)
- Imperative > Structured > Procedural—introduces more modular programming by using local variables, functions to group code (eg. using a function to abstract code)
- Declarative—"what" the result is, implicit control flow, no loops, no assignments, stateless, output depends on input, order of execution (the "how") is not the focus
- Declarative > Logic—facts, rules and goals, program infers the results, backtracking to find solutions (eg. Prolog as a language is purely declarative and logic/constraint driven)
- Declarative > Object oriented—objects have state within them, stateless between objects, inheritance, encapsulation, polymorphism
- Declarative > Event driven—asynchronous, control flow triggered by actions/events
- Declarative > Functional—similar to mathematical functions, abstracting logic as independent functions, no shared state (no side-effect), no mutable data, control flow expressed by combining functions instead of assigning values to variables, composition of functions—chain and combine, allows parallelism of computation

Example (structured) –

```
var income_m = 0, income_f = 0;

for (var i = 0; i < income_list.length; i++) {

   if (income_list[i].gender == 'M')

      income_m += income_list[i].income;

   else

      income_f += income_list[i].income;

}
```

Note:

- explicit use of **variables** that will contain the **state** of running totals;
- explicit **loop** over the data, **modifying** the variable and hence the state;
- **conditionals** to **choose the code path** at each iteration.

Example (declarative) –

```
select gender, sum(income)

from income_list

group by gender;
```

Note:

- memory cells to contain running totals are implied by the output you declare you want;
- any loop the CPU will need to perform (eg. over the income_list table) is implied by the output you declare you want and by the structure of the source data;
- conditionals (eg. case in SQL) are used in a **functional** way to s**pecify the output value** you want based on the input values, not to choose a code path.

4.3.2.7.4 How should WE paradigm?

These paradigms are not mutually exclusive. You will find yourself using principles from multiple paradigms at any given point.

Procedural paradigm is useful at the bottom / unit / low level. You'd rather use it when describing a basic code block since its the most expanded / obvious way the logic would work and the execution order is clear without over-killing on the structuring.

Declarative programming is more abstract and helps better structure the logic of complex and big applications. Its easier to see relationships and logic flow since the purpose is to abstract out the "how" to focus on the "what".

Whenever we are modelling a stated-system, or a mapping, we make use of the principles of event-based paradigm. eg: switch statements, state machines

Functional paradigm is a great way to structure code as independent, purpose-defined code blocks. By reducing dependencies on global variables / states, and allowing for functions to chain and compose together, one is able to write bug free and modularised code. This helps in building services and helps testing at the function / purpose level.

Sometimes, you want to model a service / object / user for which you might find it convenient to have states and inter-connected functionality. This is when object-oriented paradigm comes handy.

Hence, you could be modelling a system / service in an OOP, writing the code blocks there in a FP and utilising PP to write functions and code blocks. Or maybe create a large event-driven system that in its definition is the EBP.

Conclusion :

- We spend hours to understand the code compared to actual changes
- We need to leverage SOA (similar to OOP at some level) to talk in Services
- The code complexity and dealing with biz-logic is where breakage happens
- This is because we write too much of Procedural code
- We need to leverage Functional programming for code structuring

4.3.2.8 Spring 5

The Spring Web MVC framework provides Model-View-Controller (MVC) architecture and ready components that can be used to develop flexible and loosely coupled web applications. The MVC pattern results in separating the different aspects of the application (input logic, business logic, and UI logic), while providing a loose coupling between these elements. Models are responsible for encapsulating the application data. The Views render response to the user with the help of the model object. Controllers are responsible for receiving the request from the user and calling the back-end services.

The figure below shows the flow of request in the Spring MVC Framework.

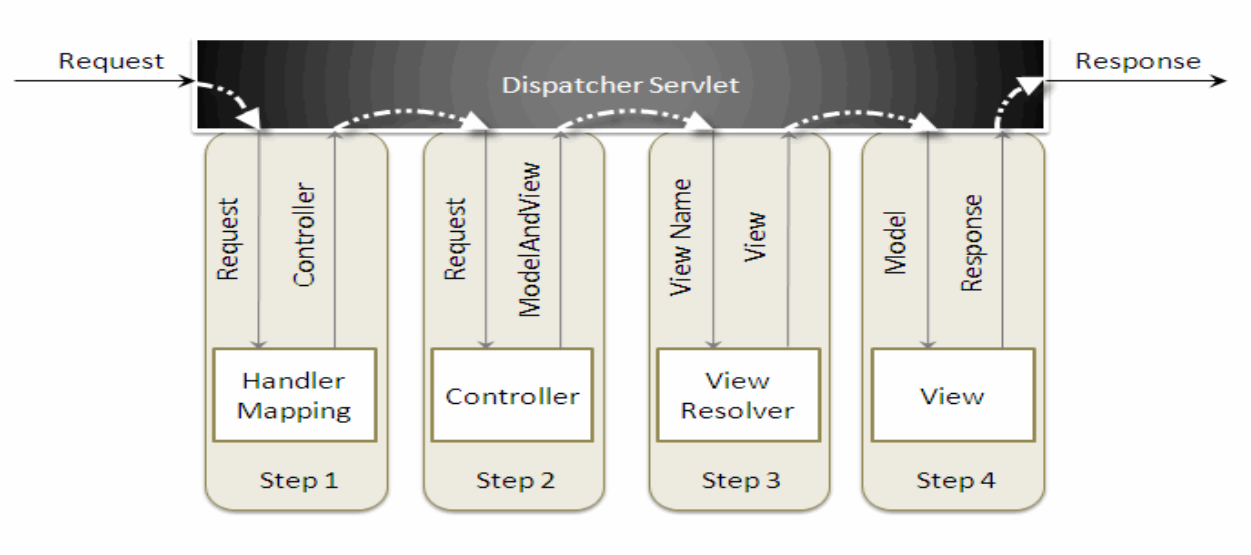


Fig-8: Flow of Spring MVC

4.3.2.9 MyBatis(against Hibernate)

MyBatis is an open source, lightweight, persistence framework. It is an alternative to JDBC and Hibernate. It automates the mapping between SQL databases and objects in Java, .NET, and Ruby on Rails. The mappings are decoupled from the application logic by packaging the SQL statements in XML configuration files. It abstracts almost all of the JDBC code, and reduces the burden of setting of parameters manually and retrieving the results. It provides a simple API to interact with the database. It also provides support for custom SQL, stored procedures and advanced mappings.

4.3.2.10 Apache Tomcat® 9

Apache Tomcat, often referred to as Tomcat Server, is an open-source Java Servlet Container developed by the Apache Software Foundation (ASF). Tomcat implements several Java EE specifications including Java Servlet, JavaServer Pages (JSP), Java EL, and WebSocket, and provides a "pure Java" HTTP web server environment in which Java code can run.

4.3.2.11 Jython (Python Interpreter)

Jython is an open source implementation of the Python programming language, integrated with the Java platform. A programmer can compile Python source code to Java bytecode, and run the code on any Java virtual machine. The integration of Python and Java gives programmers access to all Java libraries. Developers can create applets, use JavaBeans, and also designate the classes of either language as subclasses within the other. Jython itself is written in Java and supplemented by the dynamic object-oriented programming features of Python; it allows Python to run on any Java platform.

4.3.2.12 Security --> Secure API (JSON Web Tokens)

JSON Web Token (JWT) is an open standard (RFC 7519) that defines a compact and self-contained way for securely transmitting information between parties as a JSON object. This information can be verified and trusted because it is digitally signed. (still in development by Google)

- Compact: Because of their smaller size, JWTs can be sent through a URL, POST parameter, or inside an HTTP header. Additionally, the smaller size means transmission is fast.
- E.g.: vendor appllication - credit card - send JSON data - easy for humans to read & write
- openly communicates - encrypt - JSON token (with time)
- JSON is used in REST API.

5. FUTURE ENHANCEMENT

- Successful Businesses Are Powered By Artificial Intelligence

With this incredible amount of data on everything from human behavior, social issues, health informatics and machine performance, we have a huge opportunity to uncover insights on how to build better products and make people smarter. This is where machine learning and AI come in. There is simply too much data being produced and at such a high volume that it's impossible for humans to sift through it and derive intelligence from it. But with artificial intelligence computers are to digest and learn from this data and ultimately become smarter.

Additionally, computer processing power has grown at an incredible rate while the cost of processing this data has decreased significantly, making AI more accessible. In fact, AI is everywhere, in nearly every app and device that we use every day. Apple's Siri leverages natural language processing to recognize voice commands. Facebook's deep learning facial recognition algorithm can instantly identify a person with nearly 98 percent accuracy. And Amazon, Netflix and Spotify all utilize machine learning to understand how each item in their massive catalogs not only relate to one another, but also each customer's preferences.

Artificial intelligence is increasingly becoming a de facto requirement, whether it be a consumer app or enterprise system.

Every business needs to be smart about its interactions with customers and be able to provide fast, smart and more personalized engagement at every touchpoint. AI does just that.

With AI embedded into business processes, companies are suddenly armed with deeper insights on their customers and actionable recommendations on a best path forward so users across sales, service, marketing and IT can make smarter decisions.

For example, cable providers can leverage AI to improve customer retention. So if a customer has a service issue, the service agent can look at the type of issue the customer is experiencing and the client's lifetime value to recommend an appropriate action such as a credit for a streaming movie or a free technician visit for a service disruption. Or a retailer can identify new store locations based on historical data such as sales, demographics and distance from competitors and nearby events.

The next generation customer experience is powered by AI. Customer experience is also at the heart of digital business transformation. In the age of the customer, where everyone and everything is connected, more informed and decisively agile, every enterprise app must be a smart app, and all businesses must leverage AI to stay competitive and relevant. The barriers of entry for advanced analytics and AI have been lowered, democratizing intelligence in every industry and every business of all sizes.

- Business Success Depends On Smart (Not Just Big) Data

Companies are heavily investing in acquiring and developing talent, technology and business processes aimed at collecting and analyzing massive amounts of data, so that they can develop actionable business insights aimed at bolstering customer value. The key driver for digital business transformation is improving the ability to convert data to knowledge and understanding that leads to meaningful and timely action.

Five Steps to Get from Raw Data to Actionable Insights:

1. Start with determining the business problem you're trying to solve - As a starting point in dealing with all of this data, Wu recommends that businesses begin with a business problem. Collecting all the data is important because you don't know what problem you may have in the future, but in order to use the data to take business action, you need to start with a problem. "If you want to have a big data initiative or a big data strategy, first identify some problems so that the data that you collected will have more immediate value," says Dr. Wu. In doing this, data will have more value and a longer shelf life, increasing the likelihood that it will prove valuable in 10 or 20 years down the line.

Once you have collected the data to solve a particular problem, then you have to see what kind of attributes or what information and what insights you can get from this data. Everyone wants to know how to improve their awareness on social media and Wu says that there is lots of data out there about people's consumption of social media or participation in social media. Marketers can actually look at that data and do some simple types of analysis so that they can maximize their social media efforts.

2. Start with descriptive analytics - In order to move raw data to information insights, Dr. Wu says that there are three classes of analytics that people can use. The first-class is called descriptive analytics, which is a summary of historical data that has been collected that is usually shown as a visual dashboard. Dr. Wu says that 80% of most of the analytics that most companies do fall into this category. "You always start

with descriptive analytics and then if you get enough data you can become more sophisticated and then you could actually build predictive analytics. And if you are more advanced then you essentially do prescriptive analytics," says Dr. Wu.

There are three general classes of analytics for data reduction and decision support:

1. Descriptive Analytics: Compute descriptive statistics to summarize the data. The majority of social analytics fall in this category.

2. Predictive Analytics: Build a statistical model that uses existing data to predict data that that we don't have. Examples of predictive analytics include trend lines, influence scoring, sentiment analysis, etc.

3. Prescriptive Analytics: Build a prescriptive model that uses not only the existing data, but also the action and feedback data to guide the decision maker to a desired outcome. Because prescriptive models must be actionable and have a feedback data stream, social analytics are rarely prescriptive.

3. Compute sentiment with predictive analytics - The simplest type of predictive analytics that everyone is familiar with is a trend line. You look at the data and then follow some trend and you can see that if you continue to follow this trend tomorrow, or in the future, it will be a certain, predictable value. Dr. Wu says that the interesting point about predictive analytics is that you don't have to just predict the future; you can actually predict things in the past as well. In this case, you are trying to use data that you have to predict data that you don't have. "Predictive analytics is really simple and it's basically what you put into a model and the output of the models tells you something that you don't already know," says Dr. Wu.

In social media, there are a couple of types of predictive analytics that people are familiar with. For example, sentiment analysis is actually predictive analytics. "With sentiment analytics, nobody actually goes out and reports that their sentiment is positive for Apple or Android or whatever. In their natural language, they just say 'I love my iPhone' or 'I love my new android'. Using the natural language as the known data, we build a model that uses linguistic processing, so when people use this type of language; it typically means that they have positive sentiment or negative sentiment. So the sentiment is actually computed and has not actually been measured," says Dr. Wu.

4. Meet KPIs with prescriptive analytics - The simplest example of a type of prescriptive analytics is a Google map; it has prescribed a route for you to get to where you want to go. Like with predictive analytics, once you have a model you can predict things in the past. With prescriptive analytics, can prescribe what you need to do and what you need to focus on in order to get to a particular business key performance indicators (KPI), such as achieving the highest customer satisfaction or the greatest lift in revenue.

5. Get to actionable results - Wu thinks that whether it is descriptive, predictive or prescriptive, the ultimate goal is to help business decision makers to take action on the analytics of their data. "Action-ability is really important. A lot of people say that they provide actionable analytics, but what do they actually mean? Actionable is a type of analytics and is also prescriptive analytics; it tells you a course of action where you can take the action and affect the outcome," says Dr. Wu. If you cannot take an action, then it's not prescriptive analytics.

Dr. Wu explains that with prescriptive analytics there is this notion of what we call a predictive window, which means that within this window the error that you make in prediction is still acceptable. When we talk about action-ability, you have to have another measure called reaction time, which is the time that it takes for you to take action on what you have learned from these predictions. "One of the most important criteria for action-ability is that your reaction time has to be shorter than the predictive window," says Dr. Wu.

6. CONCLUSION

Thus, MyTribe is for individuals who are in need of fast and trustable services, with access to internet. With the use of technology, they can reserve professionals online. The impact of its work shall be how relationships develop between professionals and individuals.

7. REFERENCES

[1] en.wikipedia.org

[2] https://www.udacity.com

[3] https://www.medium.com/urbanclap-engineering/archive

BIOGRAPHIES

	Anindita Bhowmik Undergraduate Student, Information Technology, SRM University, Tamil Nadu
	Veeramani R. Assistant Professor, Information Technology, SRM University, Tamil Nadu