A

PROJECT REPORT

ON

NearBy iOS Application

Submitted for The Partial Fulfillment of Course of the Degree of Master of Computer Applications
(MCA VI Semester, 2017-2018)

Submitted by
Navdeep Charan
Enroll No. 15r/0006205

Guided by
Ms. Deepika Kamboj
Assistant Professor

Department of Computer Science and Engineering,
MBM Engineering College, Faculty of Engineering,
Jai Narain Vyas University, Jodhpur (Rajasthan)

Session 2017-18

DECLARATION

I **NAVDEEP CHARAN** hereby declare that this seminar titled "**NearBy iOS Application**" is a record of original work done by me under the supervision and guidance of **Ms. DEEPIKA KAMBOJ.** I further certify that this project work has not formed the basis for the award of the Degree/Diploma/Associateship/Fellowship or similar work to any candidate of any university and no part of this report is reproduced as it is from any other source without seeking permission.

Navdeep Charan

Enroll No. 15R/0006205

Roll No. 16MCA10017

CERTIFICATE

This is to certify that the Seminar Report entitled "**NearBy iOS Application**" submitted to Department of Computer Science and Engineering, MBM Engineering College, JNV University in partial fulfillment of the requirements for the award of the degree of "**MASTER OF COMPUTER APPLICATIONS**" is the bonafied record of the work done by "**NAVDEEP CHARAN** (15r/0006205)" under my supervision and guidance.

Date:

Guided by

Ms. Deepika Kamboj

(Assistant Professor)

Acknowledgement

The satisfaction that accompanies the successful completion of this project would be incomplete without mentioning the people whose ceaseless cooperation made it possible, whose constant guidance and encouragement crown all efforts with success.

I am grateful to my seminar guide **Ms. DEEPIKA KAMBOJ**, Department of Computer Science and Engineering, M.B.M. Engineering College, for the guidance, inspiration and constructive suggestions that was helpful in the preparation of this project report.

I wish to express my sincere gratitude to **Prof. Dr. Anil Gupta**, Head of Department, Department of Computer Science and Engineering, M.B.M. Engineering College, for providing me an opportunity to present my project.

I have tried my level best to make this project report error free, but I regret for errors, if any.

TABLE OF CONTENTS

Chapter 1
Company Profile

1.1 Mithra Media Pvt. Limited

1.1.1 PHILOSOPHY

By merging marketing, consulting, development, and design, our simple philosophy is to making marketing more playful and interactive, where develop marketing tools to turn passive observer into a participant. We enable our customers improve and grow their digital presence without the complication, jargon and cost of other providers. Our business philosophy is based on: Ethical business practices, we are also never evil. Always keep the entrepreneurial spirit All of our employees are first loyal to their first customers We are open and innovation company, but we also not afraid of saying no.

1.1.2 STRATEGY

Mithra Media is a place where technology is driven by creativity and the free exchange of ideas. Where creativity is balanced by a rigorous ethic of testing and analysis, where talented people of high integrity like to come to work each day. Our strategy of creating excellence in every area of operations of our customers is through: creating chic and unique design, our designer knows how to perfect the mobile experience. creating scalable solutions, our aim is to optimize our customers' ROI in any marketing campaign. delivering solutions on time, time is money and we value both.

1.2 Products and Services

Mithra Media provides following products and services:

1.2.1 MOBILE DESIGN

While there are specific rules, guidelines and specifications that Apple, Android, etc. would like to see used in your apps, at Mithra Ltd. we make sure that your mobile application is always Unique Reflects yours cooperate identity Provides maximum user experience and most importantly appealing to your targeted audience. Our goal at Mithra Media is to rapidly develop high-quality mobile applications using eye catching design practices. Over 10 years of software experience has refined our design methodologies. Adopting elements of both Waterfall and Agile development methods, Mithra Media implements a unique development process to deliver a first- class product predictably without sacrificing quality or speed. Design is as much about function, interaction, and driving engagement as it is making things visually appealing. We co-locate our visual designers, user researchers, and interaction designers to create apps that are not only visually stunning, but optimized for real world scenarios.

1.2.2 IOS SOLUTION

We develop cutting-edge applications for many of the world's leading brands and with our highly competent team, we are able to develop most complicated applications Native and web applications Integration into existing infrastructure Secure, robust solutions Collaboration with in-house designers iOS-optimized designs Scalable, resolution-independent Interfaces from strategy to launch, our end-to-end tailor made applications that are designed to make an impact.

iOS is one of the world's most advanced mobile platforms, redefining what can be done with a mobile phone. For developers, the iOS SDK and Xcode tools make it possible to create revolutionary applications that will set the bar for the next generation of mobile applications.

The iOS SDK (formerly iPhone SDK) is a software development kit developed by Apple Inc. and released in February 2008 to develop native applications for iOS. The iOS platform is supported by Apple mobile devices such as iPhone, iPod Touch and iPad. iOS is based on the C programming language.

1.2.3 ANDROID SOLUTION

With our experienced mobile application team, we enable you to realize significant business benefits from promoting your business directly to your customers, suppliers and partners. Native and web applications Integration into existing infrastructure Secure, robust solutions Collaboration with in-house designers' Android-optimized designs Scalable, resolution-independent Interfaces. Our mobile application development team has mastered sophisticated Android application development for the niche markets. Android is the leading smartphone platform in the world, with over 190 million devices in use and growing by more than 550,000 activations a day. We assist our clients penetrate this market, giving them the power to offer capabilities that could extend much further than simple day-to-day communication.

1.2.4 MOBILE WEB SOLUTION

Our experience web developers manipulate HTML5, CSS3, and JavaScript to achieve mobile browsing experiences. World Class Designers Senior level creative directors and project managers W3 validation for Coding Compatibility with range of mobile devices We understand it is important that your website represent your vision and organization in a manner consistent with your individual and corporate identity. We can assure that the data structure of the website supports search engine friendly protocol, hence the website looks as good as it works. By fashioning sites in HTML5, our mobile web engineers can rival native software expectations for the iPhone, iPad, Android, and BlackBerry. Our developed web application simply be designed with flexibility in mind to look beautiful on any combination of screen sizes and resolutions

Chapter 2
Project Overview

2.1 NearBy: Project Overview

NearBy is a Mobile Application which is specially built for Apple iPhone. The application is totally built in Swift language which is used for preparing iOS applications.

2.2 Scope of Project

NearBy project is mainly built to achieve the functionality of knowing and finding the places around you. The geographical data fetched using Google Maps must be organized in a specific way to help the user. This task can be obtained by accessing the Google Maps SDK that includes Nearby Search API and Places API in the application. Moreover, the application should be able to help the user to get reviews and photos of the place.

2.3 System Specification

The System Specification for this application give below:

2.3.1 SOFTWARE REQUIREMENTS

- OS: Apple iOS 7+

2.3.2 HARDWARE REQUIREMENTS

- Phone: Apple iPhone 7 or latest

2.4 Modules of Project

NearBy App is basically divided into four modules.

2.4.1 LOGIN OR SIGN-UP MODULE

Login or Sign-up module is specifically meant for giving app entry permission to user

2.4.2 CITY MODULE

This module lets users select the city or detect the current GPS location to Nearby search.

2.4.3 HOME MODULE

Home module is meant for displaying the categories of places to select Nearby places.

2.4.4 SEARCH MODULE

Search Module displays the around places after picking a category of selected city or current location.

2.4.5 BOOKMARK MODULE

Bookmark module is meant for showing locally saved, bookmarked places and to remove places from the bookmark list.

2.4.6 PROFILE MODULE

This module lets the user to set his/her profile information and allow to change city or location.

2.4.7 DETAILS MODULE

Details Module provides information of specific place which has three tabs:

- About
- Reviews
- Gallery

Chapter 3
Basics of Software Development

3.1 Basics of Software Development

There are number of steps in the design and development. I will first give an overview using an analogy. All the phases are not needed every time and it depends on the application. All the phases are explained below.

3.1.1 REQUIREMENTS DETERMINATION

The first stage in the design is to specify the needs or requirements. During this phase the requirements to be met by the system being developed are determined. For. E.g. If the pathology lab wants a complete computerized lab, then all the transactions and their priorities should be considered. The result of this phase is a set of requirements and their priorities.

3.1.2 REQUIREMENTS SPECIFICATION

During this phase, the application where a computer can be employed are listed. It is extremely important to get the users involved right from the initial stages of the development of the project and seriously considered their suggestions. As output of this phase an understandable plan of what the system will provide and detailed specifications of information to be provided.

3.1.3 FEASIBILITY ANALYSIS

Having the rough specifications now, the next step is to check whether it is feasible to implement or not. A Feasibility study takes into account various constraints within which the software should be implemented and operated. All the estimated resources such as human, computers, time, etc. are taken into account and compared with available resources.

3.1.4 SYSTEM SPECIFICATION ANALYSIS

Functional specifications based on revised user requirements and feasibility analyses are obtained. At the end of this phase, functional specifications, budget schedule, physical requirements are maintained.

3.1.5 HARDWARE STUDY

Based on the finalized specifications it is necessary to determine the configuration of hardware and support software essential to execute the specified application. This is based on volumes of data to be processed, frequency of reports, etc.

3.1.6 SYSTEM DESIGN

The next step is to develop the logical design of the system. The inputs to the system design are functional specifications of the system and details about the computer configuration. During this phase, the logic of the program is designed, file and databases are designed, program test plans and the implementation plans are drawn up. The system design should begin from the objectives of the system, namely, the information requirements of users, and use this to find the necessary database.

3.1.7 SYSTEM IMPLEMENTATION

During this phase writing programs, creating databases, testing programs and operation plans, documenting software, training users, data conversion if needed, installing software, trial of software and parallel runs with existing system and tests to accept software. The outputs of this phase are programs, databases, user manuals and operational manuals.

3.1.8 SYSTEM EVALUATION

After the software has been in operation for a reasonable period, it is evaluated and a plan for its improvement is drawn up.

3.1.9 SYSTEM MODIFICATION/MAINTENANCE

Modification will definitely cost time and money, but user expects modification to be made as same “Software” itself implies it soft and hence changeable. Good software would change and would design the system in a flexible way to easily accommodate changes. The output of this phase is improved system.

Chapter 4
System Study & Requirements

4.1 System Study

4.1.1 PROPOSING A NEW SYSTEM

While proposing a new system, we need to take care of many requirements of the user. The phase of Requirement determination has to be taken very seriously. As the system is being proposed for the first time, Requirement determination, Requirement specification has to be done from the beginning. All the initial stages and the future enhancement option has to be thought of.

4.1.2 FEATURES OF A NEW SYSTEM

Features of the new system includes solution to the all the problems of the user. All the requirements of the user must get converted to features of a new system. In our case it was the user requirement, that they wanted us to make an iOS app for Nearby Places Search. The application should provide functionality of download, upload, creating folders inside application directory as well as server.

4.2 System Requirement Determination

4.2.1 REQUIREMENT ANTICIPATION

The general meaning of the Requirement Anticipation is to foresee the requirements of the client in later stages of development. This can be treated as future enhancement options. As the client is going to have a brand new system, the options has to be made available to client for future enhancements.

4.2.2 ANALYSIS FOR THE REQUIREMENTS

At the heart of system analysis is a detailed understanding of all important facts of the business area under investigation. This is a very important phase of the system development as this is the phase in which the system needed by the users, the whole business process is properly understood and according to the requirements gathered during this phase, the system is developed so the proper analysis of the system is very necessary.

All the requirements are gathered by analyzing the existing system and understanding problems faced by this system. The study of reports, documentation, manuals are done. These data are used to identify the features of the system must have, including both the information of the system should produce and operational features such as processing controls, response times, and input and output methods.

Main requirements for all functionalities provided by this system are understood properly so that the business flow and user perception is clear. Also needs of users, those are not provided by this system are also got from users.

4.3 System Requirement Specification

4.3.1 FUNCTIONAL REQUIREMENTS

Whenever any new software application is being built up, the primary steps for starting is to get the functional requirements in detailed form. This phase of obtaining the pre-requisite requirements in full detail is called Functional requirement. The functional requirement would usually include various kinds of modules such as Login, Registration, User Management, obtaining the various kinds of output such as printing forms, or obtaining data in one form or other form. All such things make up smaller or bigger kinds of modules.

4.3.2 NON-FUNCTIONAL REQUIREMENTS

Nonfunctional requirements are basically those requirements which are not playing any kind of functional role in the application. These kinds of requirements make the project easy to use and facilitates the speed of transactions made by the application. Use of various kinds of GUI controls enhances the usability of the application.

4.3.3 PROCESS MODEL

Any application which has to be created requires a definite kind of process model. These process models determine the productivity of the application. Selection of proper process model is responsible for faster software development.

Using spiral model, software is developed in a series of incremental release. During early iteration, the incremental release was paper model or prototype. During later iterations, increasingly more complete versions of the engineering system were produced. Using spiral model, the software was developed in a series of incremental releases or submissions (fulfilling milestones). With later interactions, we were able to add complexities and exceptional cases.

In this project, we first started with requirements specifications then we developed the prototype and then developed and added sophistication and complexity to it.

The spiral model is a realistic approach to the development of large-scale systems. The spiral model, also known as the spiral lifecycle model, is a systems development lifecycle (SDLC) model used in information technology (IT). This model of development combines the features of the prototyping model and the waterfall model. The spiral model is favored for large, expensive, and complicated projects.

The steps in the spiral model can be generalized as follows:

- The new system requirements are defined in as much detail as possible. This usually involves interviewing a number of users representing all the external or internal users and other aspects of the existing system.
- A preliminary design is created for the new system.
- A first prototype of the new system is constructed from the preliminary design. This is usually a scaled-down system, and represents an approximation of the characteristics of the final product.
- A second prototype is evolved by a fourfold procedure: (1) evaluating the first prototype in terms of its strengths, weaknesses, and risks; (2) defining the requirements of the second prototype; (3) planning and designing the second prototype; (4) constructing and testing the second prototype.
- At the customer's option, the entire project can be aborted if the risk is deemed too great. Risk factors might involve development cost overruns, operating-cost miscalculation, or any other factor that could, in the customer's judgment, result in a less-than-satisfactory final product.
- The existing prototype is evaluated in the same manner as was the previous prototype, and, if necessary, another prototype is developed from it according to the fourfold procedure outlined above.
- The preceding steps are iterated until the customer is satisfied that the refined prototype represents the final product desired.
- The final system is constructed, based on the refined prototype.
- The final system is thoroughly evaluated and tested. Routine maintenance is carried out on a continuing basis to prevent large-scale failures and to minimize downtime.

4.4 E-R Diagram

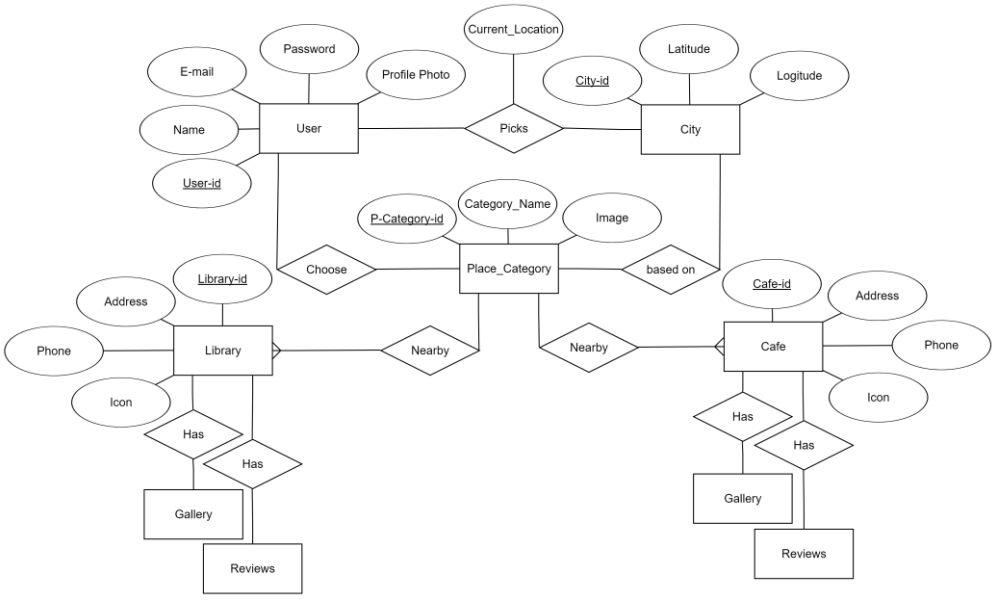

Fig. 4.4.1: E-R Diagram

Chapter 5
Feasibility Study

5.1 Feasibility Study

A feasibility study's main goal is to assess the economic viability of the proposed business. The feasibility study needs to answer the question: "Does the idea make economic sense?" The study should provide a thorough analysis of the business opportunity, including a look at all the possible roadblocks that may stand in the way of the cooperative's success. The outcome of the feasibility study will indicate whether or not to proceed with the proposed venture. If the results of the feasibility study are positive, then the cooperative can proceed to develop a business plan.

A feasibility study should examine three main areas:

- market issues
- technical and organizational requirements
- financial overview

5.1.1 MARKET ISSUES

The primary area that the feasibility study needs to address is potential market opportunities for the cooperative. If an adequate level of demand does not exist for the product and the COMPANY does not know how to differentiate its product so that it can compete with established industry players, then the proposed venture should not be pursued.

Questions that need to be answered in this area of the feasibility study include:

- What type of industry is the COMPANY planning to enter? What are its primary features?
- What are the possible target markets for the COMPANY's product? What demographic characteristics do they possess? How large are these markets? Where are they located? Is the market expected to grow in the future?
- Will the COMPANY be competing in a mature industry or a growth industry?
- Who are the COMPANY's competitors in this market? How large are these competitors? How established are they? How do they price their goods? How will these competitors react to the entrance of the COMPANY?
- How will the COMPANY differentiate its product from those of its competitors? What are the competitors' strengths and weaknesses, and how would the COMPANY compare against them? How does the COMPANY plan on gaining market share?
- What is the projected market share for the COMPANY?

Data that can help to answer these questions may be found in already-published information or through primary research activities such as market surveys conducted on behalf of the COMPANY. Relevant information may be found through various sources such as government statistical publications, trade journals, industry reports, or companies such as Dun & Bradstreet. The Internet has also opened up new routes to obtaining information. The answers to market-related questions should help the COMPANY develop realistic estimates of the projected demand for the COMPANY's product for the first several years of operation. Based on this projected demand, the COMPANY can determine its anticipated level of business volume, which is needed in order to design the processing facilities. If the projected business volume is not large enough to justify a processing facility, then the project is not feasible.

5.1.2 TECHNOLOGICAL AND ORGANIZATIONAL REQUIREMENTS

This area concerns the internal set-up of the cooperative. Questions to be answered in this area include:

Plant and equipment issues:

- What type of equipment and technology will the business need to produce its product? What are the costs involved? This includes both the initial purchase and installation costs of the equipment as well as the operational costs of running the equipment.
- Who are the potential suppliers of this equipment? Where are they located? What sort of service and warranties do they provide? How long will it take to acquire the equipment and begin operations?
- Based on its projected business volume, how much raw product will be required by the COMPANY? What are the quality specifications? Will the COMPANY have a sufficient membership base that can provide the raw materials?
- What are the possible locations for the COMPANY's facility? What size of facility is needed? What are the costs of the building? Does the proposed location have adequate access to infrastructures and services such as major highways, railways, and utilities? Will the COMPANY build its own facility, or purchase an existing location?
- Where will the facility be located relative to the COMPANY's customers? Who will be responsible for the transportation of goods between the facility and the market? What are the transportation costs involved?

Managerial and organizational issues:

- Is the COMPANY organizational structure the right one for this business? How important are delivery contracts and a fixed source of supply to the success of the business?
- What qualifications are needed to manage these operations? What are the key staff positions that need to be filled?
- What type of experience should management have? Are there potential candidates available to fill such positions? What will be the cost factor involved in finding and retaining acceptable candidates?

5.1.3 FINANCIAL OVERVIEW

Based on the estimates that have been gathered from the preceding sections of the study, the COMPANY needs to determine its overall financial situation. Sources and uses of financing should be listed. Questions such as the following need to be considered:

- What are the total start-up costs required in order to begin operations? For instance, what are the capital costs of the land, plant and equipment, and other start-up costs such as legal and accounting costs?
- What are the operating costs involved? These include the daily costs involved in running the business, such as wages, rent, utilities, and interest payments on outstanding debt. These will determine the cash flow requirements of the COMPANY.
- Based on the estimated demand, what are the COMPANY's revenue projections? How will the COMPANY determine its pricing arrangements?
- What are the possible sources of financing for the COMPANY? Who are potential lenders? What will be their required terms and limitations of borrowing?
- Based on the estimated revenues and costs, what is the projected profit(loss) of the COMPANY? What is the break-even point?

Chapter 6
Project Planning, Scheduling and Tracking

6.1 Project Management

Project management is a carefully planned and organized effort to accomplish a successful project. A project is a one-time effort that produces a specific result, for example, a building or a major new computer system. This is in contrast to a program, which is 1) an ongoing process, such as a quality control program, or 2) an activity to manage a series of multiple projects together. In some countries, the term "program" refers to a software tool and the term "programme" can mean a TV or radio show.

Project management includes developing a project plan, which includes defining and confirming the project goals and objectives, identifying tasks and how goals will be achieved, quantifying the resources needed, and determining budgets and timelines for completion. It also includes managing the implementation of the project plan, along with operating regular 'controls' to ensure that there is accurate and objective information on 'performance' relative to the plan, and the mechanisms to implement recovery actions where necessary.

6.2 Project Planning

Project planning is a discipline for stating how to complete a project within a certain timeframe, usually with defined stages, and with designated resources. One view of project planning divides the activity into:

- Setting objectives (these should be measurable)
- Identifying deliverables
- Planning the schedule
- Making supporting plans

Supporting plans may include those related to: human resources, communication methods, and risk management.

Computer hardware and software project planning within an enterprise is often done using a project planning guide that describes the process that the enterprise feels has been successful in the past.

Tools popularly used for the scheduling part of a plan include the Gantt chart and the PERT chart.

6.3 Project Scheduling

Scheduling is an inexact process in that it tries to predict the future. While it is not possible to know with certainty how long a project will take, there are techniques that can increase your likelihood of being close. If you are close in your planning and estimating, you can manage the project to achieve the schedule by accelerating some efforts or modifying approaches to meet required deadlines.

One key ingredient in the scheduling process is experience in the project area; another is experience with scheduling in general. In every industry area there will be a body of knowledge that associates the accomplishment of known work efforts with a time duration. In some industries, there are books recording industry standards for use by cost and schedule estimators. Interviewing those who have had experience with similar projects is the best way to determine how long things will really take.

When preparing a schedule estimate, consider that transition between activities often takes time. Organizations or resources outside your direct control may not share your sense of schedule urgency, and their work may take longer to complete. Beware of all external dependency relationships. Uncertain resources of talent, equipment, or data will likely result in extending the project schedule.

Experience teaches that things usually take longer than we think they will, and that giving away schedule margin in the planning phase is a sure way to ensure a highly stressed project effort. People tend to be optimistic in estimating schedules and, on average, estimate only 80% of the time actually required.

Failure to meet schedule goals is most often due to unrealistic deadlines, passive project execution, unforeseen problems, or things overlooked in the plan.

6.3.1 THE GANTT CHART

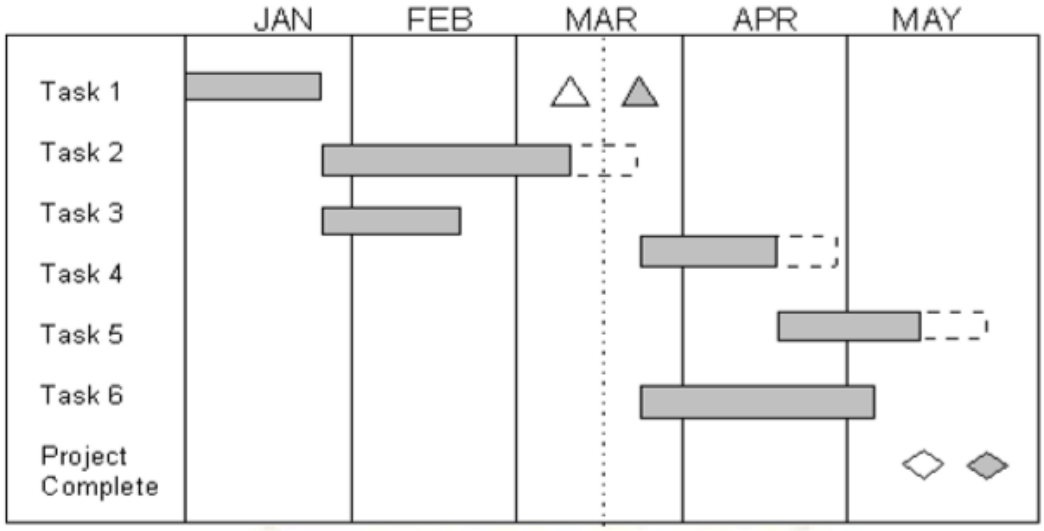

Fig. 6.3.1.1: The Gantt Chart

Taking its name from early project management innovator Henry L. Gantt, the basic Gantt chart is an easy way to document schedules. It is a horizontal-bar schedule showing activity start, duration, and completion. It shows the connection between events and the calendar, and provides a graphical analog of the activity duration.

The Gantt schedule can illustrate the relationship between work activities having duration, events without duration that indicate a significant completion, and milestones that represent major achievements or decision points. Various annotations can be used to communicate the progress of the project effort compared to the baseline plan, as well to depict in a graphical way areas where there are modified expectations from the baseline plan.

Once a Gantt schedule has been established for a project, progress should be periodically plotted against the baseline schedule. If different functional areas are involved in a project, each area may need its own detailed schedules to support the project master schedule. In such cases it is important that working schedules be linked to a common master schedule in a way that they can be easily updated. Each activity or event on the schedule should have a responsible individual assigned, so there is clear ownership and so schedule status can be updated without a lot of fuss.

6.3.2 RESOURCE LEVELING

Projects will often be confronted by time and organizational constraints that limit their ability to obtain human resources. Sometimes staff can be supplemented through temporary help from technical service agencies. When staffing requirements are identified and constraints are understood, work plans can sometimes be adjusted to fit requirements to available resources.

Resource scheduling is one of the greatest challenges for projects without access to large organizational or job-market resource pools. Project planning should address such issues as redundancy of critical resources, resource capacity, bench strength in vital areas, and contingency plans to handle departures of key personnel.

Most of the popular project management software packages enable the project resource planner to assign staff to project tasks, display resource requirements profiles, and adjust the schedule of slack tasks so resource requirements more closely fit those available in the organization. Some packages can display multiple project resource requirements to facilitate organization-wide resource management, optimization, and leveling. Individual project requirements may be adjusted by manipulating schedule slack in tasks not on the critical path. This can facilitate allocation and leveling of staff throughout the organization.

Unless one person is working on each task full time, the schedule duration on the Gantt chart will not be the same as the effort required. Effort requirements will drive project cost, but durations will drive the schedule. These distinctions are helpful when reconciling project and resource schedules.

6.3.3 "CRASHING" THE SCHEDULE

Efforts to accelerate a project schedule are commonly grouped under the term "crashing" the schedule. Maybe this term was coined to suggest that there is always some price for driving a project to completion sooner than normal. There are a number of ways to improve the schedule when your boss says, I need it sooner!

- Add people to the schedule. Additional staff must be added early in a project or they will slow it down while learning the ropes. If you add people, you may also need to add staff for supervision and coordination, so staff are fully applied.
- Improve productivity and work longer hours. A good team atmosphere with management support can help make this happen. Without positive nourishment of this process, you could lose your team to attrition.
- Review schedule dependencies and look for opportunities to overlap tasks or make serial tasks concurrent or parallel activities. This requires greater coordination and sometimes involves increased risks which need to be managed carefully.
- Review the project scope and remove or delay features or functionality from the project critical path.
- Consider innovative approaches such as a different development methodology, alternative technologies, or out-sourcing options.

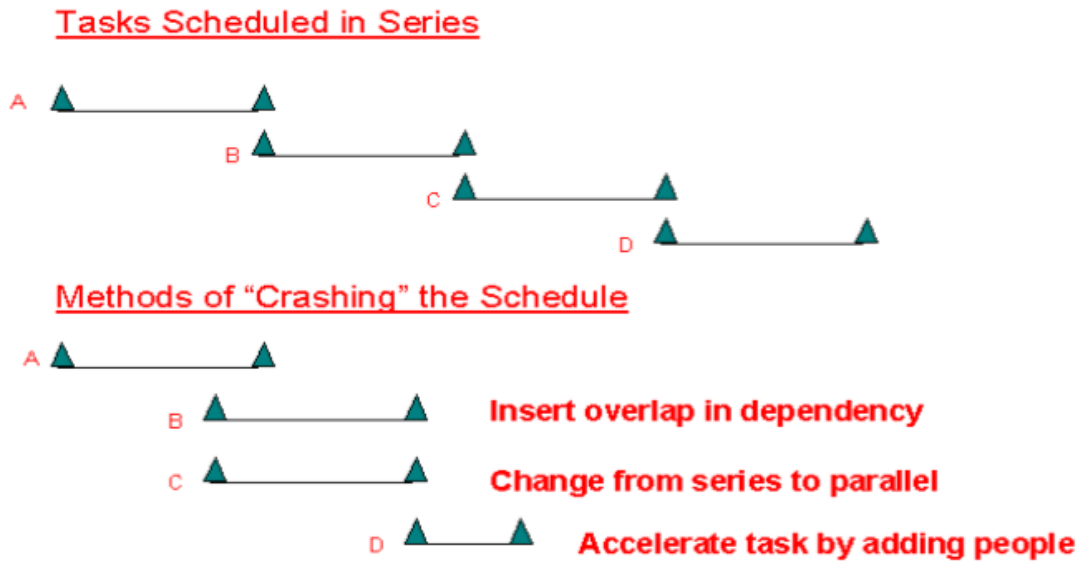

Fig. 6.3.3.2: "Crashing" The Schedule

6.4 Risk Management

Risk management is an important part of planning for businesses. The process of risk management is designed to reduce or eliminate the risk of certain kinds of events happening or having an impact on the business

6.4.1 DEFINITION OF RISK MANAGEMENT

Risk management is a process for identifying, assessing, and prioritizing risks of different kinds. Once the risks are identified, the risk manager will create a plan to minimize or eliminate the impact of negative events. A variety of strategies is available, depending on the type of risk and the type of business. There are a number of risk management standards, including those developed by the Project Management Institute, the International Organization for Standardization (ISO), the National Institute of Science and Technology, and actuarial societies.

6.4.2 TYPES OF RISK

There are many different types of risk that risk management plans can mitigate. Common risks include things like accidents in the workplace or fires, tornadoes, earthquakes, and other natural disasters. It can also include legal risks like fraud, theft, and sexual harassment lawsuits. Risks can also relate to business practices, uncertainty in financial markets, failures in projects, credit risks, or the security and storage of data and records.

6.4.3 GOALS OF RISK MANAGEMENT

The idea behind using risk management practices is to protect businesses from being vulnerable. Many business risk management plans may focus on keeping the company viable and reducing financial risks. However, risk management is also designed to protect the employees, customers, and general public from negative events like fires or acts of terrorism that may affect them. Risk management practices are also about preserving the physical facilities, data, records, and physical assets a company owns or uses.

6.4.4 PROCESS FOR IDENTIFYING AND MANAGING RISK

While a variety of different strategies can mitigate or eliminate risk, the process for identifying and managing the risk is fairly standard and consists of five basic steps. First, threats or risks are identified. Second, the vulnerability of key assets like information to the identified threats is assessed. Next, the risk manager must determine the expected consequences of specific threats to assets. The last two steps in the process are to figure out ways to reduce risks and then prioritize the risk management procedures based on their importance.

6.4.5 STRATEGIES FOR MANAGING RISK

There are as many different types of strategies for managing risk as there are types of risks. These break down into four main categories. Risk can be managed by accepting the consequences of a risk and budgeting for it. Another strategy is to transfer the risk to another party by insuring against a particular, like fire or a slip-and-fall accident. Closing down a particular high-risk area of a business can avoid risk. Finally, the manager can reduce the risk's negative effects, for instance, by installing sprinklers for fires or instituting a back-up plan for data.

Chapter 7
System Implementation

7.1 Implementation Environment

For the implementation of the project, the environment required for MAC OS X 10.6 +. I was provided with MAC OS 11.31 High Sierra which is required for the XCODE, the development IDE for the implementation of cocoa based projects.

7.1.1 MAC OS HIGH SIERRA

New technologies at the heart of the system make your Mac more reliable, capable and responsive and lay the foundation for future innovations. macOS High Sierra also refines the features and apps you use every day. It's macOS at its highest level yet.

macOS High Sierra introduces new core technologies that improve the most important functions of your Mac. From re-architecting how it stores your data to improving the efficiency of video streaming to unleashing the full power of your graphics processor, it's all central to today's Mac experience.

Your documents. Your photos. Your mail. Your apps. To your Mac, everything you care about is data. And a file system is what organizes all that data into files and folders you can access with a click. Our current file system was designed in the early days of Mac and it has performed beautifully ever since. But today's flash-based Mac systems open up new possibilities for innovation, so it's time to lay a new foundation. With macOS High Sierra, we're introducing the Apple File System to every Mac with all-flash internal storage, with an advanced architecture that brings a new level of security and responsiveness.

With the rising popularity of 4K video, a new industry standard arrives on Mac — HEVC (High Efficiency Video Coding, also known as H.265). It can compress video up to 40 per cent more than H.264, the current standard for video compression. Using HEVC, videos stream better and take up less space on your Mac, while preserving the same visual quality.

Navigate, organize and edit your photos like never before with new features in the Photos App. Tailor your web browsing experience in Safari. Find what you're looking for even faster in your Mail inbox. macOS High Sierra makes a great experience even greater.

macOS High Sierra introduces Hindi as a new language throughout the entire Mac system. From the menu bar to the dock to built-in apps, everything is designed to feel natural to users who speak and read Hindi.

7.2 Introduction to iOS 11

iOS 11 sets a new standard for what is already the world's most advanced mobile operating system. It makes iPhone better than before. It makes iPad more capable than ever. And now it opens up both to amazing possibilities for augmented reality in games and apps. With iOS 11, iPhone and iPad are the most powerful, personal and intelligent devices they've ever been.

iPad has always been a powerful way to work, play and learn. And iOS 11 brings it to life like never before. New features and capabilities let you get more done more quickly and easily, making your iPad experience even more powerful and personal. Do with it what you will. Because now you can.

The new Files app brings all your files together. You can easily browse, search and organize all your files in one place. And there's a dedicated place for your recent files — those on your iPad as well as in apps, on your other iOS devices, in iCloud Drive and across other services like Box, Dropbox and OneDrive.

The new Dock is a foundational change for iPad. It's now available from any screen. So with just a swipe, you can open and switch apps instantly. And you can customize it with more of your favorites. Apps you've opened recently and apps that are currently open on your iPhone or Mac appear in the right side of the Dock.

iOS 11 makes it easier and more intuitive than ever to multitask. You can open a second app directly from the Dock and both apps remain active in Slide Over as well as Split View. You can drag the second app in Slide Over to the left. And you can get back to your favorite App Spaces in the redesigned App Switcher.

iOS 11 makes Apple Pencil for iPad more versatile, powerful and natural than ever. It's more useful for all kinds of tasks, such as taking notes, getting work done or simply being creative.

iOS is the world's largest augmented reality platform. Games and apps now offer fantastically immersive and fluid experiences that go far beyond the screen. By taking advantage of the latest in AR technology, you can digitally redecorate your home, explore inside an object without taking it apart or even walk with dinosaurs. The possibilities are endless.

iOS 11 makes iPhone and iPad so smart, they learn from you. So capable, they help you in more powerful and personal ways. And so in tune with you, they do things for you before you even ask.

The App Store has been redesigned from the ground up to help you discover new apps and games you can't live without. You'll see daily stories by experts, a dedicated Games tab, lists for all kinds of apps and much more. It's the biggest thing to come to the App Store since apps.

7.3 Introduction to Swift 4

Swift is a powerful and intuitive programming language for macOS, iOS, watchOS and tvOS. Writing Swift code is interactive and fun, the syntax is concise yet expressive, and Swift includes modern features developers love. Swift code is safe by design, yet also produces software that runs lightning-fast.

Swift 4 builds on the strengths of Swift 3, delivering greater robustness and stability, providing source code compatibility with Swift 3, making improvements to the standard library, and adding features like smart key paths and serialization, all while shortening build times and reducing the size of app binaries.

In Swift Playgrounds you create small programs called "playgrounds" that instantly show the results of the code that you write. A single line of code can make amazing things happen. Interactive lessons teach key coding concepts, and additional challenges and templates encourage you to explore code in exciting new ways and to create something completely unique. There are even playgrounds that use Bluetooth to control robots, drones, and other hardware accessories. It's easy to share your creations with friends, or record and post videos of your playgrounds in action.

Swift 4 was developed in the open at Swift.org, with source code, a bug tracker, mailing lists, and regular development builds available for everyone. This broad community of developers, both inside Apple as well as hundreds of outside contributors, work together to make Swift even more amazing. Swift already supports all Apple platforms as well as Linux, with community members actively working to port to even more platforms. We're excited to see more ways in which Swift makes software safer and faster, while also making programming more fun.

Swift is the result of the latest research on programming languages, combined with decades of experience building Apple platforms. Named parameters brought forward from Objective-C are expressed in a clean syntax that makes APIs in Swift even easier to read and maintain. Inferred types make code cleaner and less prone to mistakes, while modules eliminate headers and provide namespaces. Memory is managed automatically, and you don't even need to type semi-colons. These forward-thinking concepts result in a language that is easy and fun to use.

7.3.1 FEATURES OF SWIFT 4

- Faster, easier to use Strings that retain Unicode correctness and add support for creating, using and managing substrings
- Smart key paths for type-safe, efficient, extensible key value coding for Swift types
- Enhancements to creating and manipulating Dictionary and Set types
- Extends support of archival and serialization to struct and enum types and enables type-safety for serializing to external formats such as JSON and plist
- Enforced exclusive access to memory

7.4 Google Maps SDK for iOS

With the Maps SDK for iOS, you can add maps based on Google maps data to your application. The SDK automatically handles access to the Google Maps servers, map display, and response to user gestures such as clicks and drags. You can also add markers, polylines, ground overlays and info windows to your map. These objects provide additional information for map locations, and allow user interaction with the map.

When using the SDK, you need to comply with the Google Maps Platform Terms of Service and ensure that your app complies with applicable laws. Note that when using the SDK, your application name and version, authentication information and a cross-application anonymous identifier is automatically sent with each request.

If you use the Maps SDK for iOS in your application, you must include the attribution text as part of a legal notices section in your application. Including legal notices as an independent menu item, or as part of an "About" menu item, is recommended.

You can get the attribution text by making a call to `[GMSServices openSourceLicenseInfo]`.

With the Maps SDK for iOS, you can build apps that target native 32-bit or 64-bit devices running iOS 8.0 and later. Developing an application with the Maps SDK for iOS requires at least Xcode 9.0 with a target SDK of 11.0 or later (note that setting a target SDK of 11.0 or later will not prevent your app from running on iOS 8.0).

Applications that use Maps URLs require that the target device has Google Maps for iOS installed.

Use of the Maps SDK for iOS within iOS app extensions is unlikely to work, due to the strict memory restrictions which are applied. For example, use of the SDK within a custom keyboard is unlikely to work.

7.5 Facebook Login API

The Facebook SDK in Swift enables people to sign into your app with Facebook Login. When people log into your app with Facebook they can grant permissions to your app so that you can retrieve information or perform actions on the Facebook platform on their behalf.

To add a Facebook login button to your app, add the following code snippet to a view controller.

```
import FacebookLogin

func viewDidLoad() {

   let loginButton = LoginButton(readPermissions: [ .publicProfile ])
```

```
    loginButton.center = view.center

    view.addSubview(loginButton)

}
```

Your app can only have one person logged in at a time. We represent each person logged into your app with AccessToken.current. The LoginManager sets this token for you and when it sets current it also automatically writes it to a keychain cache. The AccessToken contains userId which is used to identify the user.

You should update your view controller to check for an existing token at load. This eliminates an unnecessary app switch to Facebook if someone has already granted permissions to your app. Example:

```
func viewDidLoad() {

  if let accessToken = AccessToken.current {

    // User is logged in, use 'accessToken' here.

  }

}
```

When you add Facebook Login, your app can ask someone for permissions to access a subset of that person's data or perform actions on their behalf.

For LoginButton use the readPermissions constructor to request additional read permissions.

In your view header file add:

```
// Extend the code sample "1. Add Facebook Login Button Code"

// In your viewDidLoad method:

loginButton = LoginButton(readPermissions: [ .publicProfile, .Email, .UserFriends ])
```

Your app will now ask for the person's email address and friend list.

7.6 Google Login API

Before you can begin integrating your iOS app with the Google Sign-In components, you must download the dependencies and configure your Xcode project. The steps on this page do just that. The next steps then describe how to integrate Google Sign-In features into your app.

- Install Xcode
- Install CocoaPods

Google Sign-In uses CocoaPods to install and manage dependencies. Open a terminal window and navigate to the location of the Xcode project for your application. If you have not already created a Podfile for your application, create one now:

```
pod init
```

Open the Podfile created for your application and add the following:

```
pod 'GoogleSignIn'
```

Save the file and run:

```
pod install
```

This creates an `.xcworkspace` file for your application. Use this file for all future development on your application.

Google Sign-in requires a custom URL Scheme to be added to your project. To add the custom scheme:

- Open your project configuration: double-click the project name in the left tree view. Select your app from the **TARGETS** section, then select the **Info** tab, and expand the **URL Types** section.
- Click the + button, and add your reversed client ID as a URL scheme.

When completed, your config should look something similar to the following (but with your application-specific values):

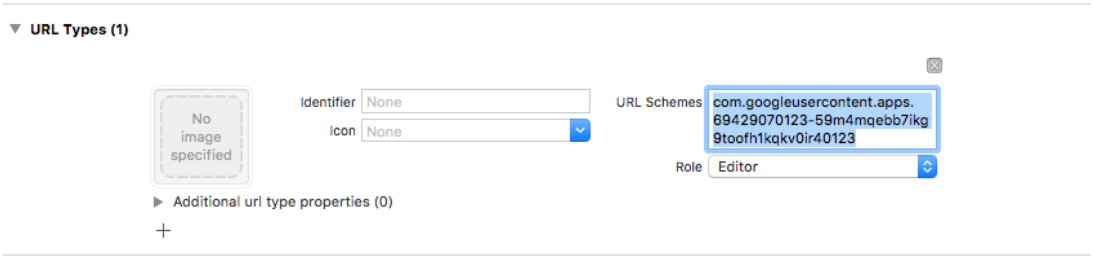

Fig. 7.6.1: Google API Login URL

Now that you've downloaded the project dependencies and configured your Xcode project, you can add Google Sign-In to your iOS app.

Chapter 8
System Testing and Quality Assurance

8.1 Introduction to Testing

Software Testing is the process of executing a program or system with the intent of finding errors. Or, it involves any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets its required results. Software is not unlike other physical processes where inputs are received and outputs are produced. Where software differs is in the manner in which it fails. Most physical systems fail in a fixed (and reasonably small) set of ways. By contrast, software can fail in many bizarre ways. Detecting all of the different failure modes for software is generally infeasible.

Unlike most physical systems, most of the defects in software are design errors, not manufacturing defects. Software does not suffer from corrosion, wear-and-tear -- generally it will not change until upgrades, or until obsolescence. So once the software is shipped, the design defects -- or bugs -- will be buried in and remain latent until activation.

Software bugs will almost always exist in any software module with moderate size: not because programmers are careless or irresponsible, but because the complexity of software is generally intractable -- and humans have only limited ability to manage complexity. It is also true that for any complex systems, design defects can never be completely ruled out.

8.2 Creating Testing Strategy for Mobile Applications

Most NoSQL databases do not perform ACID transactions, a tried and true technique for ensuring that data remains consistent across the entire database as it is moved around. Instead, NoSQL relies on the principle of "eventual consistency." This provides some performance advantages, but it poses the risk that data on one database node may go out of sync with data on another node.

Some NoSQL implementations, like FoundationDB, try to provide the best of both worlds by doing ACID-like transactions while retaining the fluidity of NoSQL design. But in general, data consistency remains a fundamental challenge for NoSQL.

The mobile device world is characterized by a plethora of devices, screen resolutions operating systems, operators and locations. In a world that is comprised of tens of thousands of different client devices and over 400 operators around the globe, testing your mobile application can become a real problem.

In this article we will make an attempt to describe the main challenges in testing your mobile applications as part of the mobile application lifecycle and offer some solutions that will help you improve your mobile testing strategy, reduce your testing costs and improve your TTM (time to market).

Main challenges in testing mobile applications:

8.2.1 SCREEN RESOLUTION

You need to make sure that your application works properly on every handset out there and on every screen resolution, in order to make sure that your customers and potential customers will have the best user experience possible. This problem gets even bigger when you take into consideration not only the common screen sizes for mobile phones (320x480, 176x220 and 240x320), but also the strengthening stream of tablet devices that are rumored to pour over the market in 2011 as well as the tablets that already in mass use (Like the iPad tablet with 1024x768, and Samsung Galaxy Tab with 1024 x 600).

8.2.2 DEVICE INPUT METHOD

Applications often require the user to interact, and in order to do so, they rely on the handset input method. Touch screens are becoming more and more popular in recently released smartphones, but there are still some very popular manufacturers, such as BlackBerry and Nokia, who offer non-touch screen-based phones. If you want to make sure that your application is suitable for everyone, you need to test it on a numeric keypad, QWERTY keypad, TrackBall device and touch screen device.

8.2.3 OPERATING SYSTEMS

You may test your application's performance on one operating system and be happy with the response time and overall performance, but different operating systems vary greatly in processing speed and memory size, a variation which results in marketing a poor performing application. Thus, you must test your mobile application at least on the main popular operating systems and make sure you have backward compatibility.

8.2.4 NETWORK LIMITATIONS

Operators determine different boundaries for their networks. For example, networks have gateways (Proxy) through which the application is communicating with the server, and this communication can be optimized by the operator in order to make the content more suitable for specific devices, resulting in changing the user experience. You need to test your mobile applications on real handsets connected to real operators, to make sure you see what your end user will see.

Proposed solutions:

Testing your mobile applications can indeed be very difficult, but there several solutions that can simplify your mobile testing strategy.

8.2.5 CHOOSE YOUR MOST IMPORTANT DEVICES

Organizations who want to test their mobile applications must plan their mobile testing strategy carefully and make an effort to minimize the amount of handsets being tested.

In order to do so, the organization should take into consideration the expected number of users for a device and/or operating system as well as the marketing needs for the application, such as target audience and market share of devices. After weighing these factors, the organization can decide which handsets are wanted to test application's functionality and design.

8.2.6 REMOTE HANDSET ACCESS

This type of mobile testing solution is comprised of real handsets, connected to real operators, which can be accessed via the web. This solution lets organizations test different devices with different screen resolutions and operating systems, saving the trouble and the costs in buying the real handsets.

8.2.7 AUTOMATION TOOLS

Automation tools let you write a testing script that can be executed on several devices or countless times on one device, thus cutting quality assurance and regression tests' costs. The right automation tool can also check the end-to-end functional flow in all possible platforms, indemnity problems as pre-defined in the script, make reports and enable collaboration between stake holders in the organization such as: developers, product managers, marketers and QA teams.

8.2.8 CONCLUSION

Mobile testing is becoming more and more complex over time as the mobile devices market gets bigger and more important. Never the less, big and small organizations are finding out they can't give up on publishing their own mobile applications, whether designated to the end user or to the corporate worker. In order to do so, organizations must think ahead of their mobile testing strategy and use existing tools, such as remote device access and automated scripting tools, in order to simplify their mobile testing.

Chapter 9
User Manual

9.1 Screen Designs

9.1.1 SPLASH SCREEN

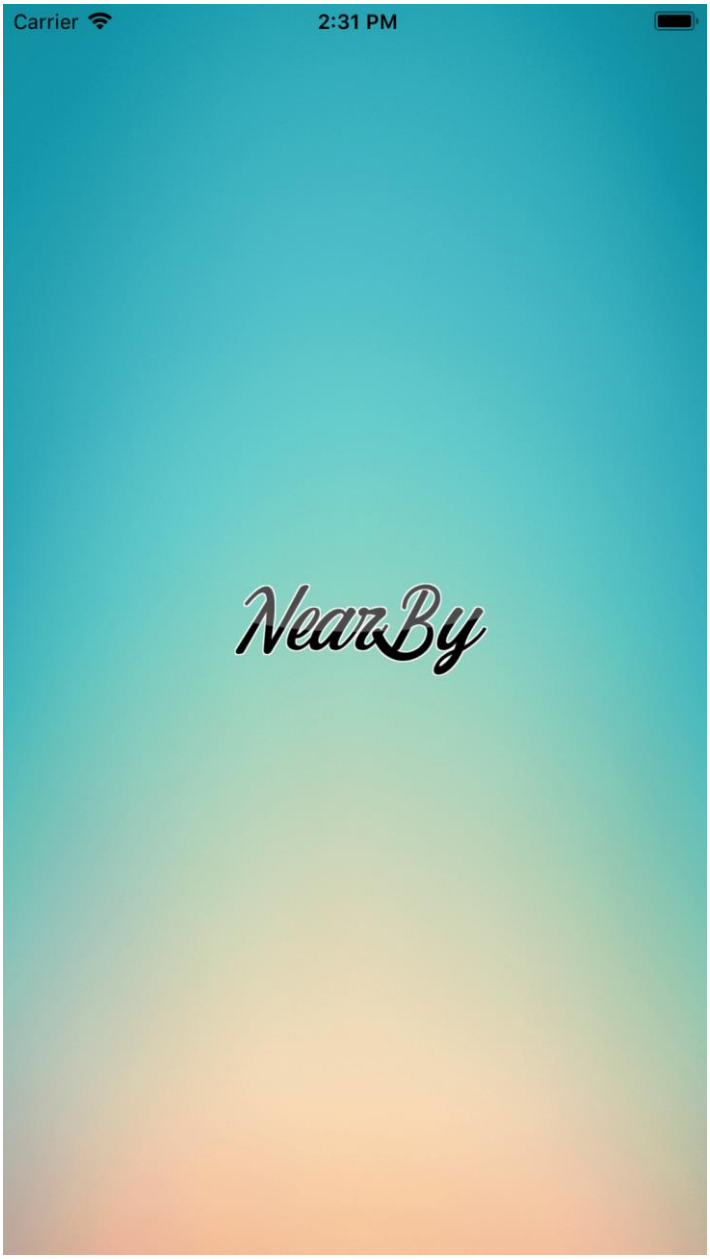

Fig. 9.1.1.1: Splash Screen

Note: This is a splash screen of NearBy App.

9.1.2 ENTRY SCREEN

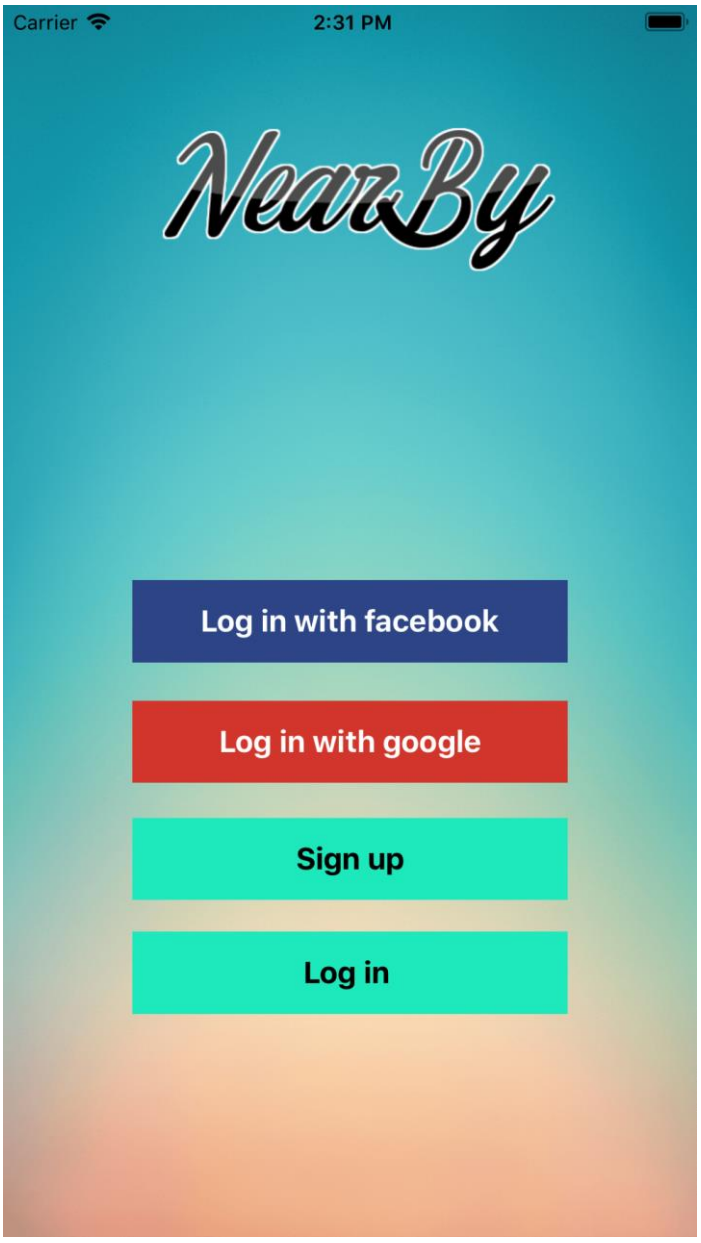

Fig. 9.1.2.1: Entry Screen

Note: This screen allows its users to Login with Facebook or Google or Login using local account and gives Sign up option for new users.

9.1.3 LOGIN & SIGN-UP

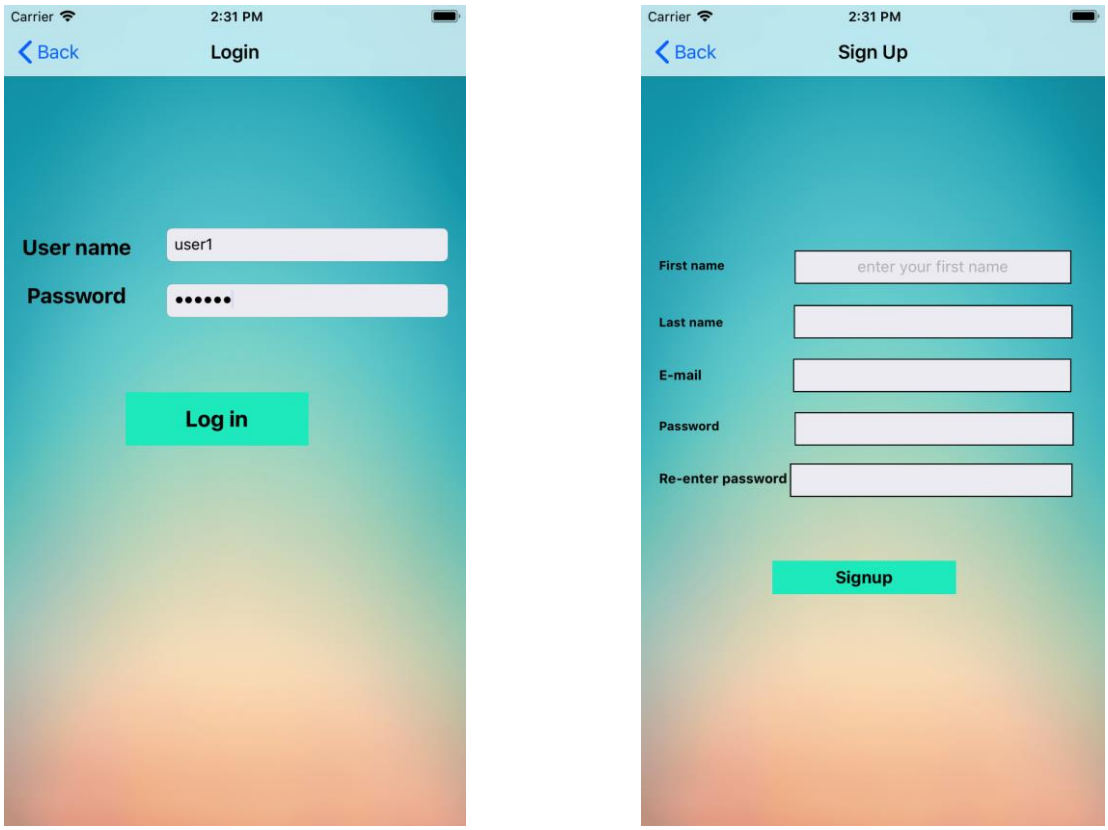

Fig. 9.1.3.1: Login & Sign-up Screen

Note: This screen enables its users to Login or Sign-up.

9.1.4 SELECT CITY

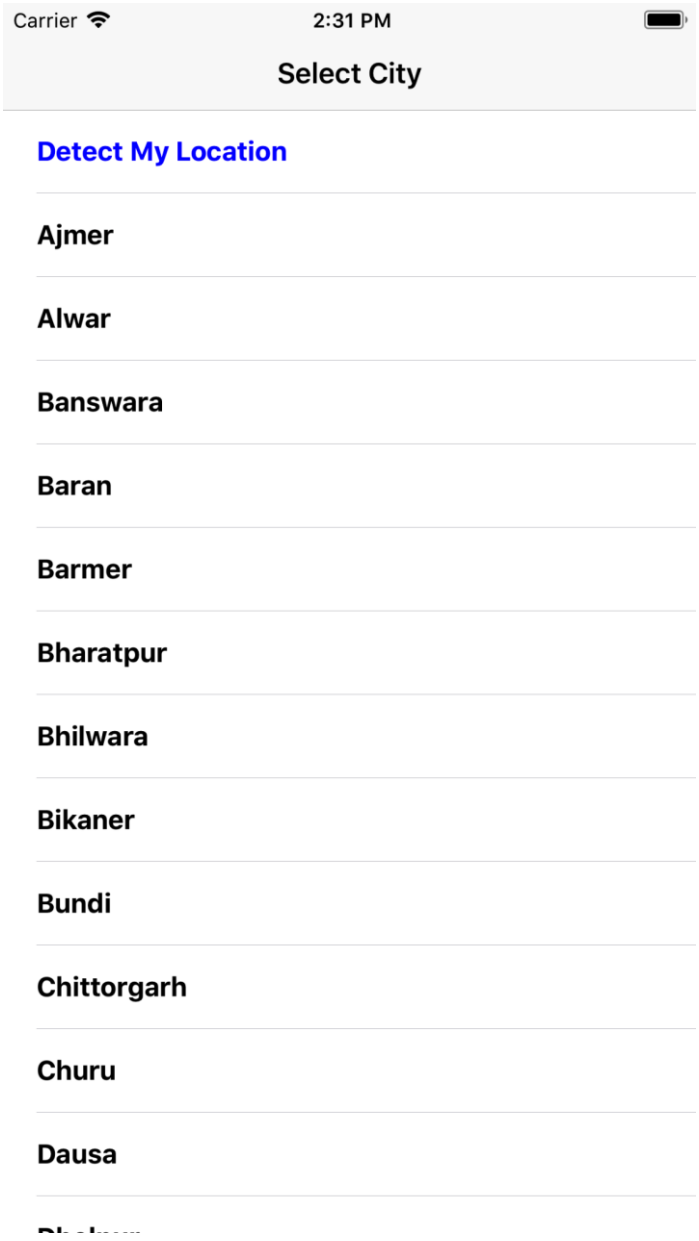

Fig. 9.1.4.1: Select City

Note: This screen let you select a city or your current location.

9.1.5 HOME SCREEN

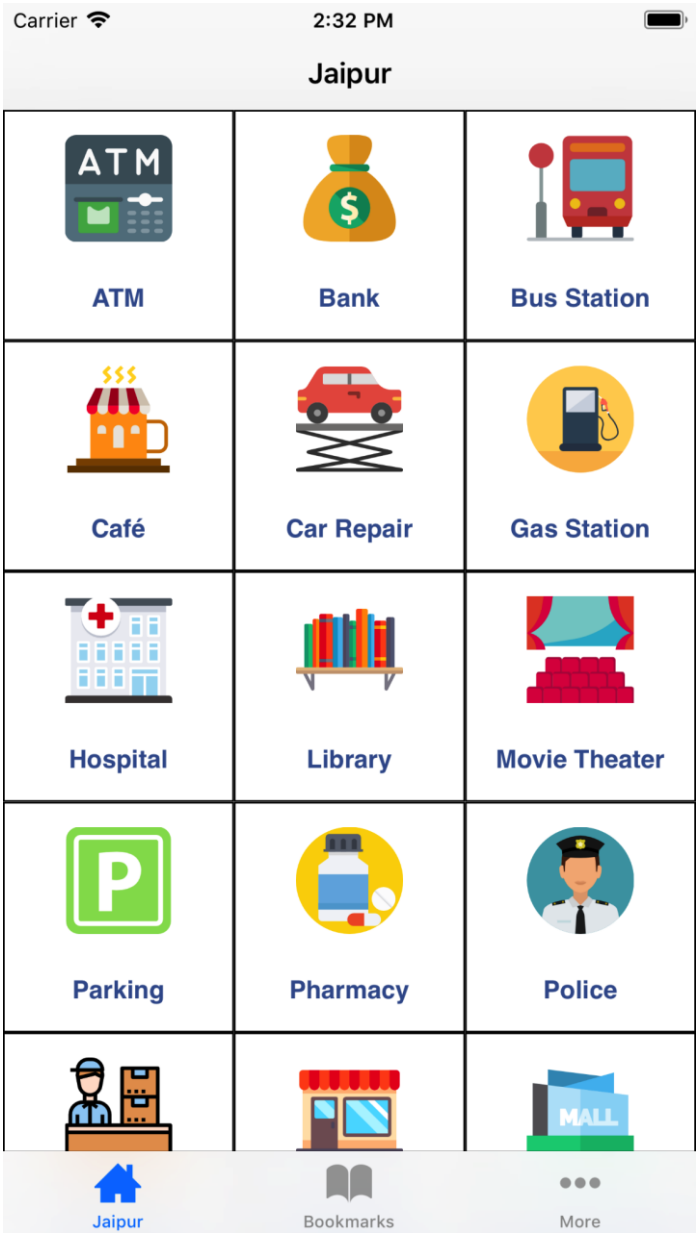

Fig. 9.1.5.1: Home Screen

Note: This screen let you choose what you want to search nearby.

9.1.6 NEARYBY SEARCH SCREEN

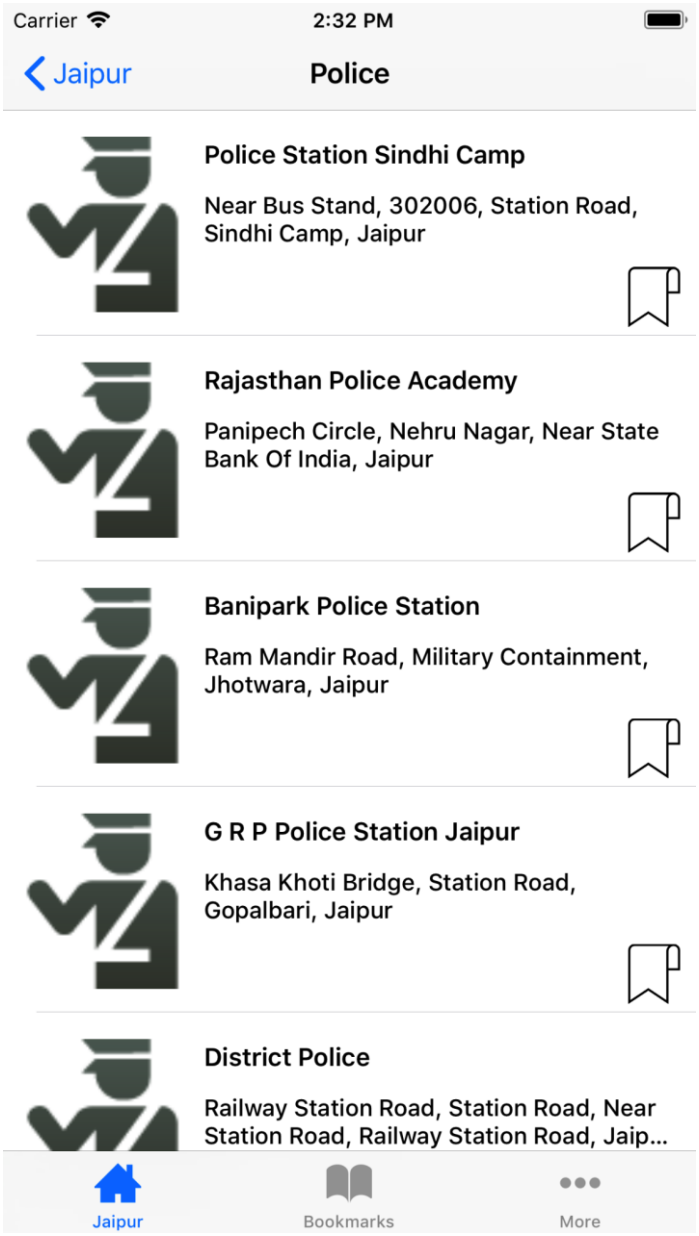

Fig. 9.1.6.1: NearBy Search Screen

Note: This screen shows the nearby places of specific category.

9.1.7 ABOUT SCREEN

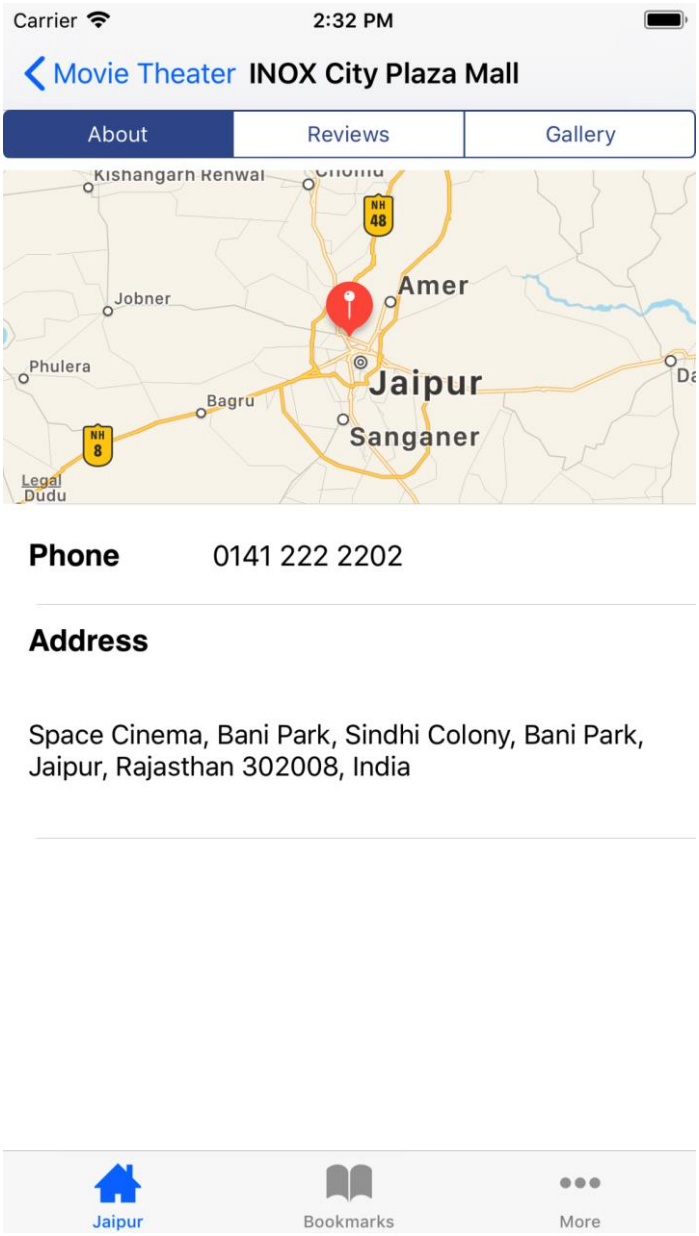

Fig. 9.1.7.1: About Screen

Note: This screen displays the general details of selected place with map.

9.1.8 REVIEWS SCREEN

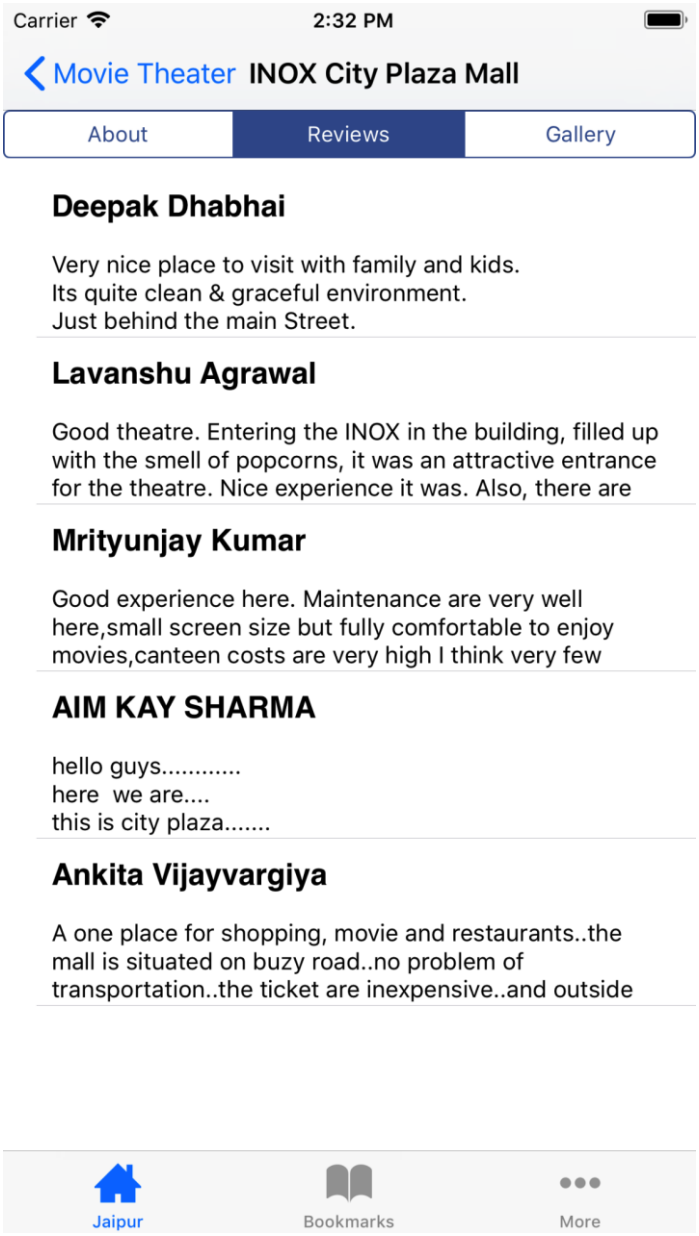

Fig. 9.1.8.1: Reviews Screen

Note: This screen displays user reviews of selected place.

9.1.9 GALLERY SCREEN

Fig. 9.1.9.1: Gallery Screen

Note: This screen displays the photos of selected place.

9.1.10 BOOKMARK SCREEN

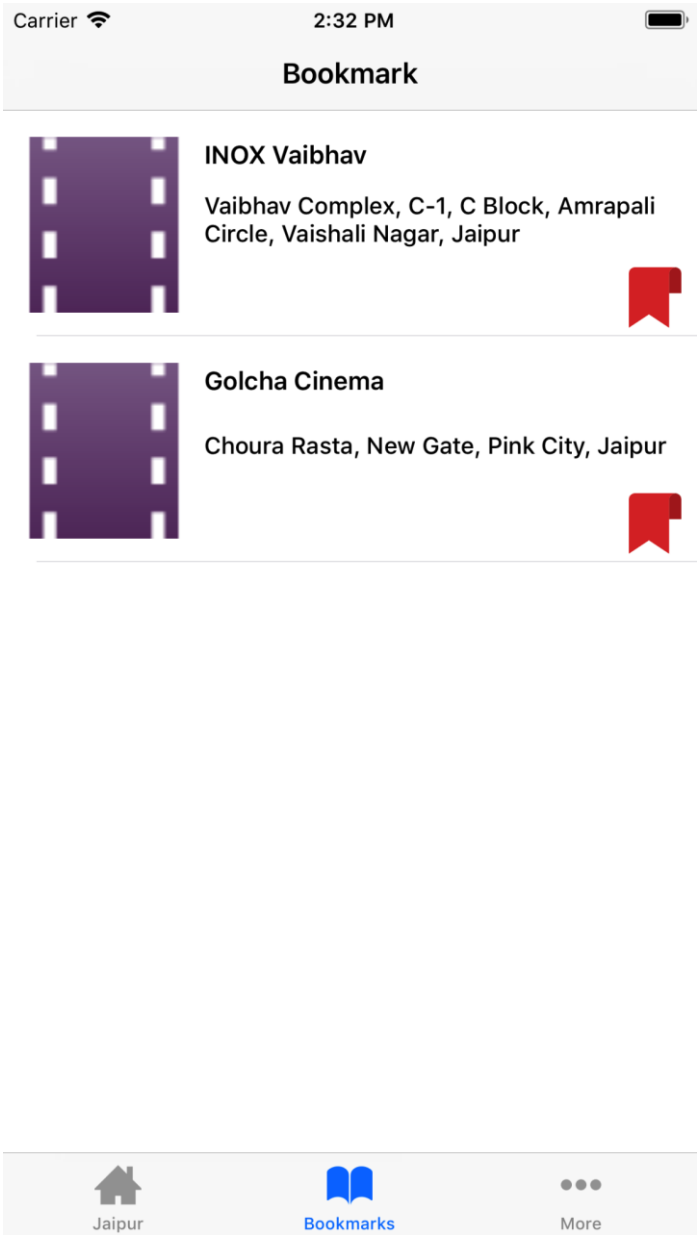

Fig. 9.1.10.1: Bookmark Screen

Note: This screen displays the bookmarked places.

9.1.11 PROFILE SCREEN

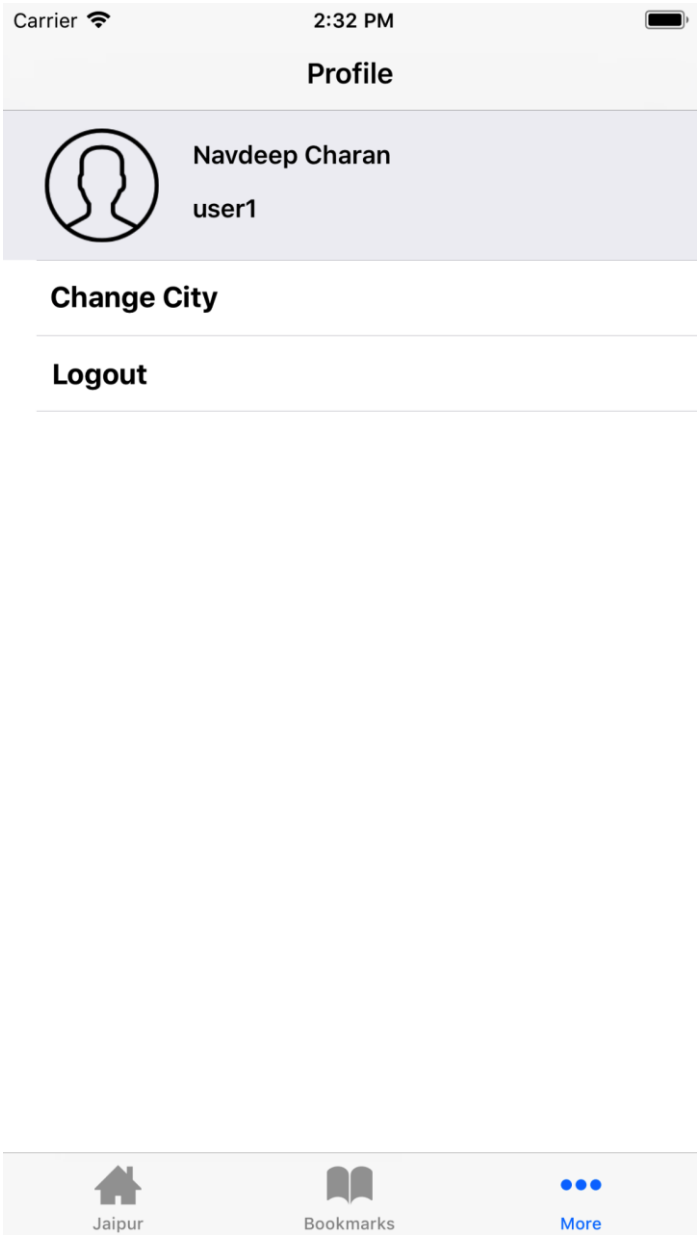

Fig. 9.1.11.1: Profile Screen

Note: This screen let you set your profile and city.

Chapter 10
Limitations and Future Enhancements

10.1 Limitations

As the project is still in its development phase, there has to be a certain limitation of the application. Few of its limitation are it doesn't enable the app to save local account on the server because there's still need for a web hosting service. Some additional important features such as sharing user visit to place on social media is missing yet.

10.2 Future Enhancements

The future enhancements of this application can be a very large versatile system. If we consider the facilities that can be provided over iOS, the possibilities are huge. But the near case future enhancement would be allowing its user in-app navigation to selected place on the map. Secondly, Web integration would be surely done in the near future too.

Conclusion & Role Statement

Conclusion

On the whole, the project has made me learn so many new as well as important things. The purpose of final semester training is fulfilled with this project. And above all, the things which I have learnt will be useful for all the upcoming projects.

Role Statement

My role as a trainee was full of learning new things, new technology, new language, new culture, new targets and of course a totally new definition. I got to learn many new things such as how to enhance programming speed of the project along with saving other important resources, how to achieve targets, etc.

During the phase of training, all my experiences were fortunately very good

References

- https://www.apple.com/in/macos/high-sierra/
- https://www.apple.com/in/ios/ios-11/
- https://developer.apple.com/swift/
- https://developers.google.com/maps/documentation/ios-sdk/intro
- https://developers.facebook.com/docs/facebook-login/ios/
- https://developers.google.com/identity/sign-in/ios/start-integrating
- http://mithramedia.co.in