Software Engineering
System Requirements Specification
For
Cindle

Prepared by:

- **Yahia Ashraf Aglan**
- **Hassan Ahmed ElDesouky**
- **Abdelrahman Mahmoud Eldeep**
- **Abdelrahman M. Ebied**
- **Ahmed Emad Mohamed**

Prepared for: SE301

Instructor: Dr.Reda Hussien

Date: October 1st, 2019

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
	26 Nov	Adding more details	0.1

1. Introduction

1.1 Purpose

This Software Requirements Specification provides a complete description of all the functions and specifications of the web application “Cindle”, The main part of this application we will develop a social interaction and sharing experiences throughout gathering different people with the same interests in real-life events where they can enjoy their hobbies and interests with alike minds away from social media.

1.2 Document Conventions

Lucidchart: site we use to create a Use case and all diagrams for the project

Term	**Definition**
IEEE	Institute of Electrical and Electronics Engineers
SRS	Software Requirements Specification
FAQ	Frequently Asked Question
OS	Operating System
REQ	Requirement

1.3 Intended Audience and Reading Suggestions

- Developers who can review the project’s capabilities and more easily understand where their efforts should be targeted to improve or add more features to it.
- Project testers can use this document as a base for their testing strategy as some bugs are easier to find using a requirements document.
- End-users of this application who wish to read about what this project can do.

1.4 Product Scope

The scope of this project is to develop a web application for building a small communities of people that share the same interests and hobbies, enhance them to develop their skills and increasing the learning curve of each one, we aim to make an optimal meeting with a comfortable place with suitable time to allow people to share interests, as initial start we will work in Alexandria, Kafr ElSheikh in Egypt.

1.5 References

[IEEE] The applicable IEEE standards are published in “IEEE Standards Collection,” 2001 edition
[Software Engineering] book we study from it for Dr.Reda Hussien
[JavaScript] is a scripting language, primarily used on the Web. It is used to enhance HTML pages and is commonly found embedded in HTML code
https://www.linkedin.com/
[Adobe xd] the tool we use to design the UI for the site
TODO: example tools for interface

2. Overall Description

2.1 Product Perspective

This is an application designed to connect people and help the users to do thier own desired acticities at the time they want and at the most suitable place that we can find. It’s also a web-based and mobile application. The application will connect people with the same interests. The mobile application will work on mobile Android and iOS devices and the web application will work on *every* modern web browser. It will have functions as selecting your interests and activities, data managing, matching and selecting the most suitable place, confirming the event details, communication with the attendees, collect feedback about the event, and finally arrange the next meetup/ event date. All of the information will be kept on a database that can be accessed by only the users who have the right to.

2.2 Product Functions

The application will consist of these main tasks:

- **Authentication**:
  . The user registration will have to create an account
  . The user login if he has an account
- **Selecting your interests and activities:**
  . The user will choose his desired interests and activities
  . The user will be able to choose multiple activities that he’s interested in doing.
- **Matching and selecting the most suitable place:**
  . After a number of people select an interest or activity.
  . Our application will match every group of people with the most suitable place to do the activity.
  . The matching will be based on the nearest available places for all of them.
- **Confirming the event details:**
  . Will show the user several choices based on the event’s place and the number of attendee.
  . The user will confirm attending the event in his desired place.

- **Communication with the attendees:**
  After the user has confirmed attending the event, a chat room will open for all of the attendees to communicate between each other.

- **Collect feedback about the event:**
  . After the event is finished a feedback screen will show up.
  . The user will submit the feedback about the event overall and the the place that the event was held in.
- **Arrange the next meetup/ event date:**
  . After submitting the feedback we will ask the user if he needs to attend the next event.

To start the user will have to create an account or login if the user has created an account previously. Then the user will choose his desired interests and activities. After a number of people select an interest or activity at the same time spot the application will match the users based on the nearest available places for all of them. Then, the user will select and confirm the most suitable place for him. After that, the user will be able to see and communicate with all of the attendees of the event. After the event is finished the attendees will be able to give feedback about the event. Finally, the application will ask the user if he needs to attend the next event.

2.3 User Classes and Characteristics

- General User: The user who wants to do/ practice his desired activity.
- Host User: The user who will be in charge about choosing the placeses in which the event will be held in.
- Admin User: The user who can control the website and has root access to it, like adding new activities, event, and banning users if they violated the terms and conditions.

2.4 Operating Environment

Firstly we will build a web application that will operate on modern web browsers and then we will build a mobile application that will operate on Android and iOS devices.

2.5 Design and Implementation Constraints

One important constraint is privacy and security. Users should be accessing only the authenticated data. The application will be based on the following technologies for the web-based version: HTML, CSS, JavaScript, Angular, NodeJS, Express. The mobile application for the iOS version will be based on Swift and the Android version will be based on Java. The database used for our application is MySQL. The management tools that will be used is Trello. Finally, we will use Git as version control technology for the application.

2.6 User Documentation

- User Manual: To help users navigate their way through the application.
- FAQ: Answers to the most asked questions.

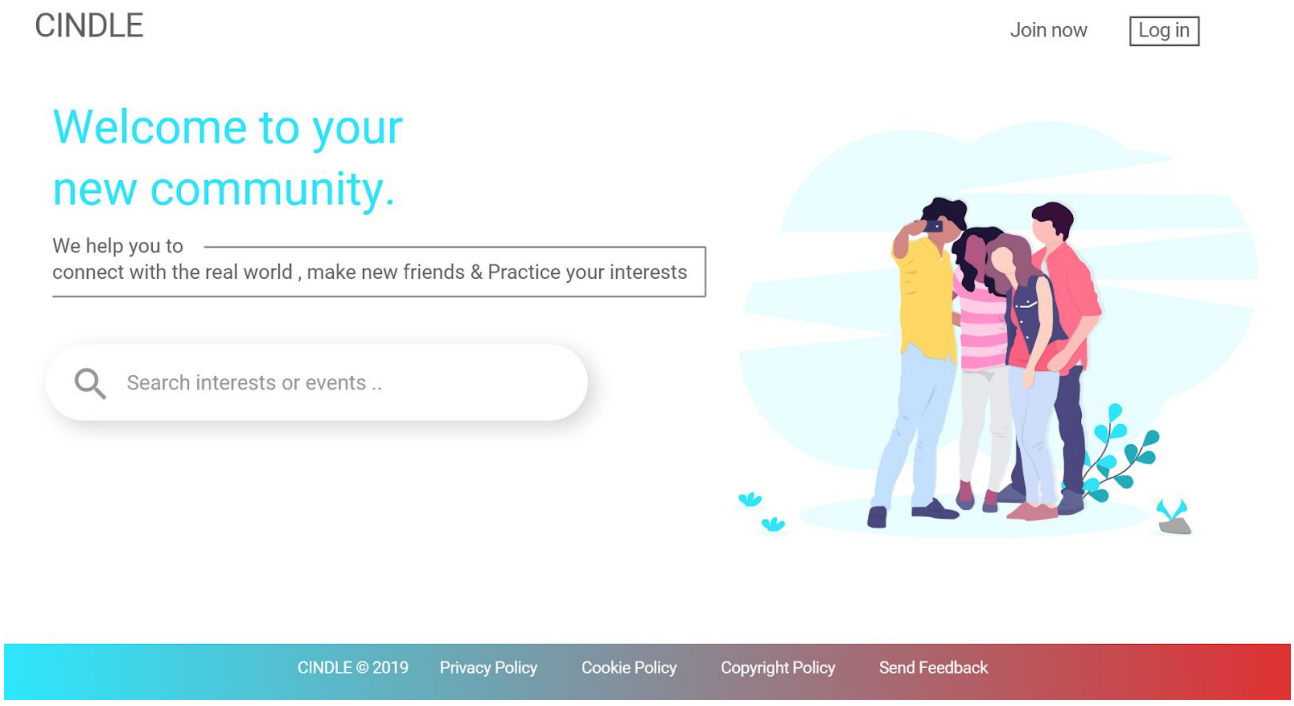
2.7 Assumptions and Dependencies

Users of this application are any Android, iOS or the web browser that loads this application in their device. All of the users are in the same class, only one type of user exits. The operating environment is, as just mentioned above, is a mobile or a computer device. A mobile or a computer device that can support basic dependencies of the application is expected for proper user experience. On the other hand, our database server and services can operate on any OS like Windows or Ubuntu that can supply the database server's fundamental dependencies and needs

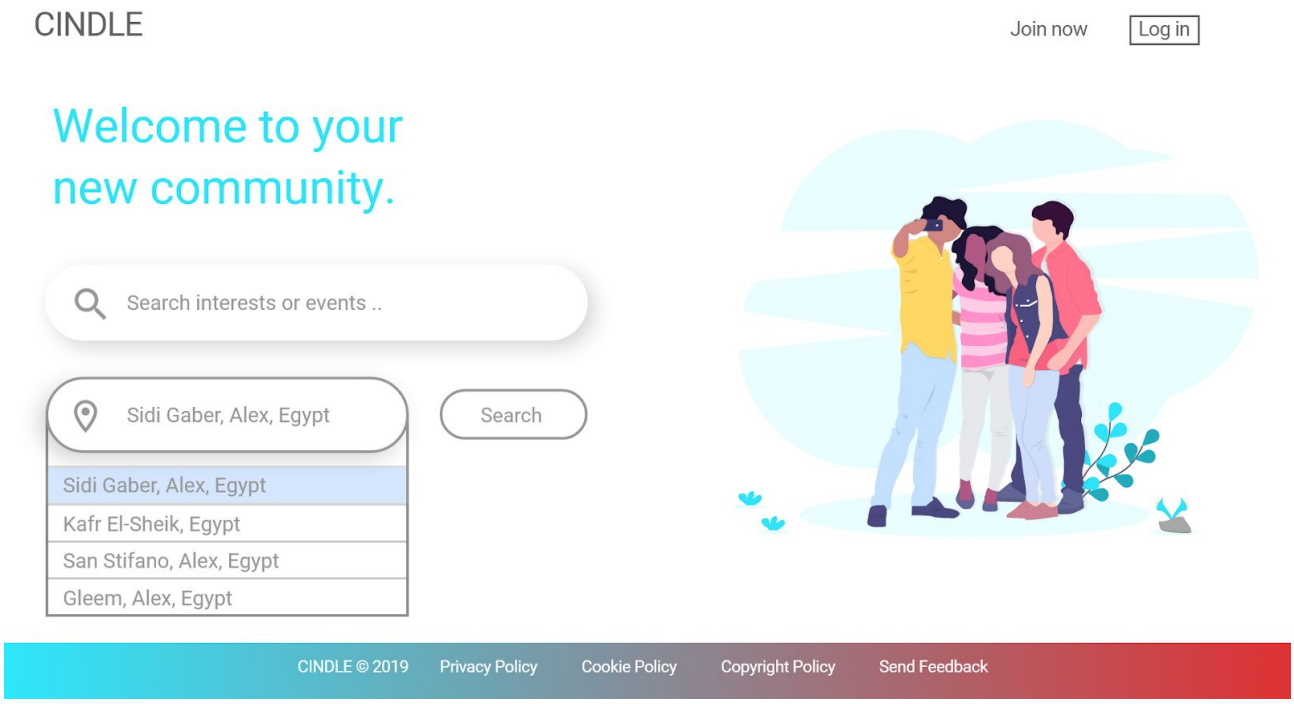
3. External Interface Requirements

3.1 User Interfaces

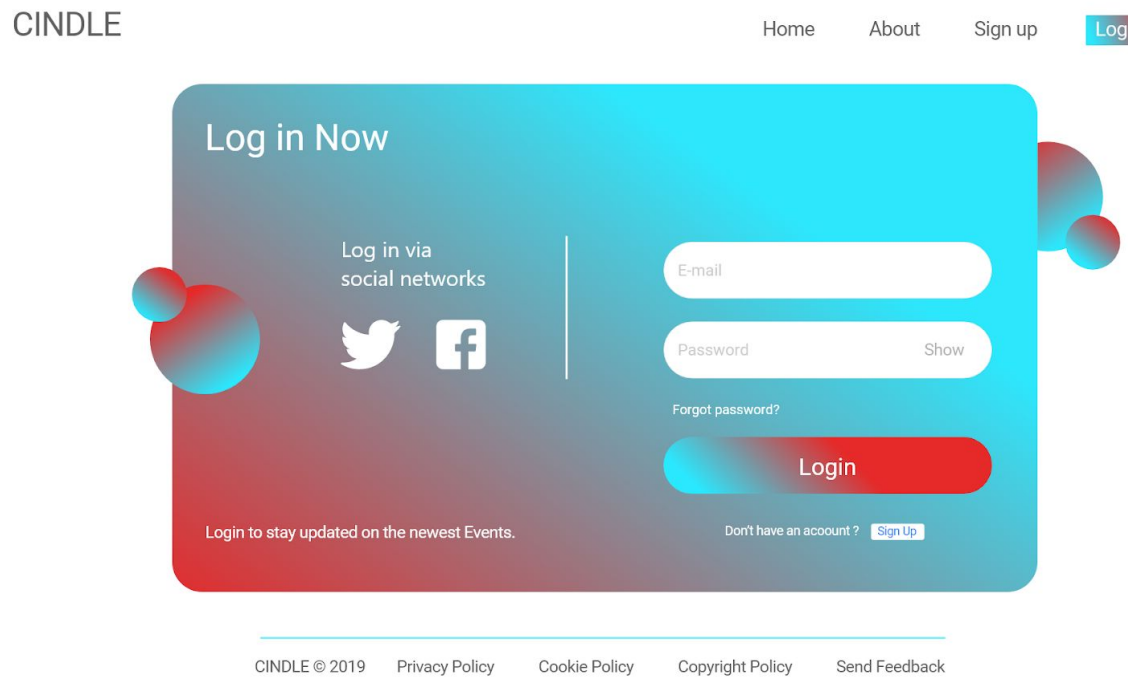
1. Cindle's Home & search for events Screen:



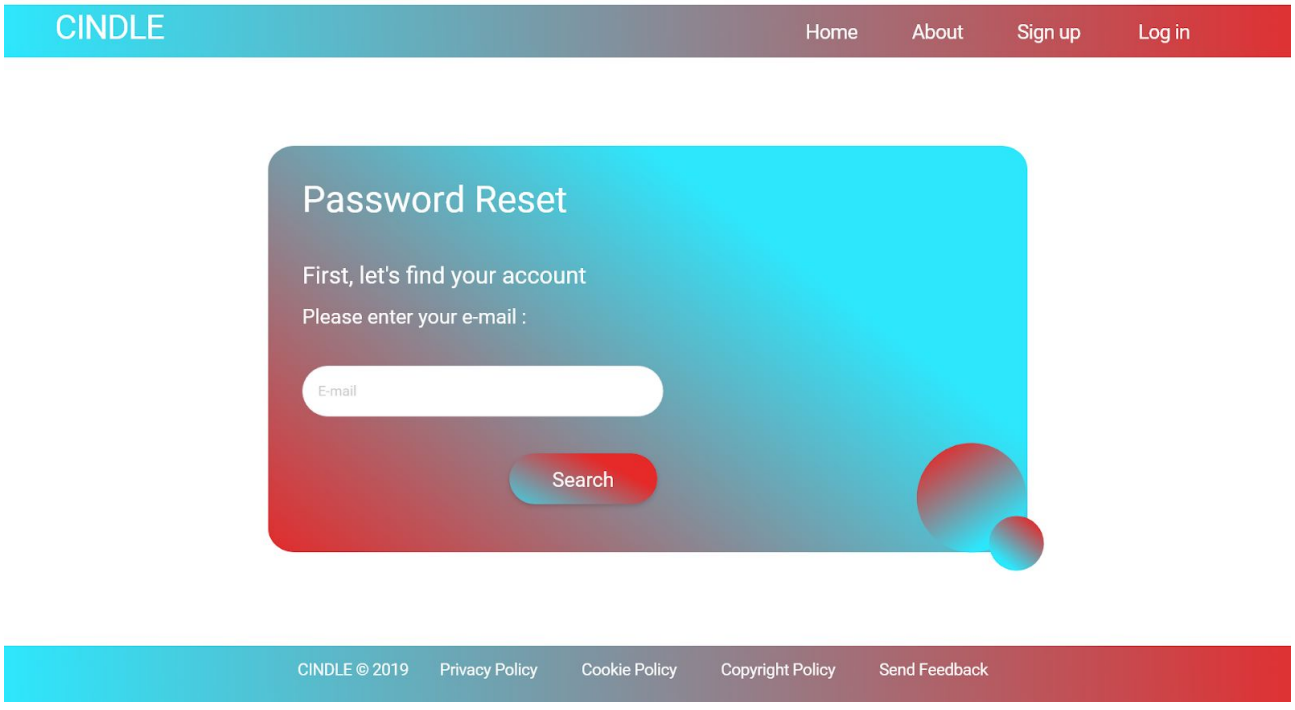
2. Cindle's Home & search with location Screen:



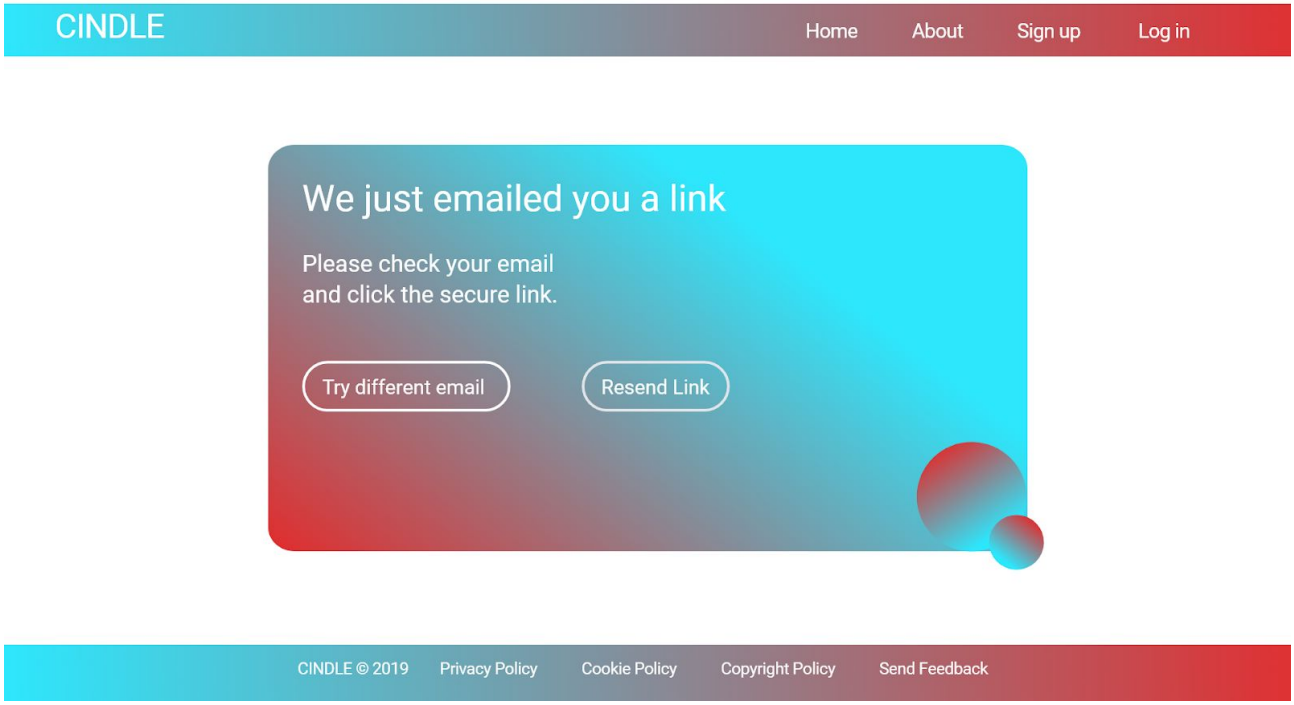
3. Cindle's Login Screen:



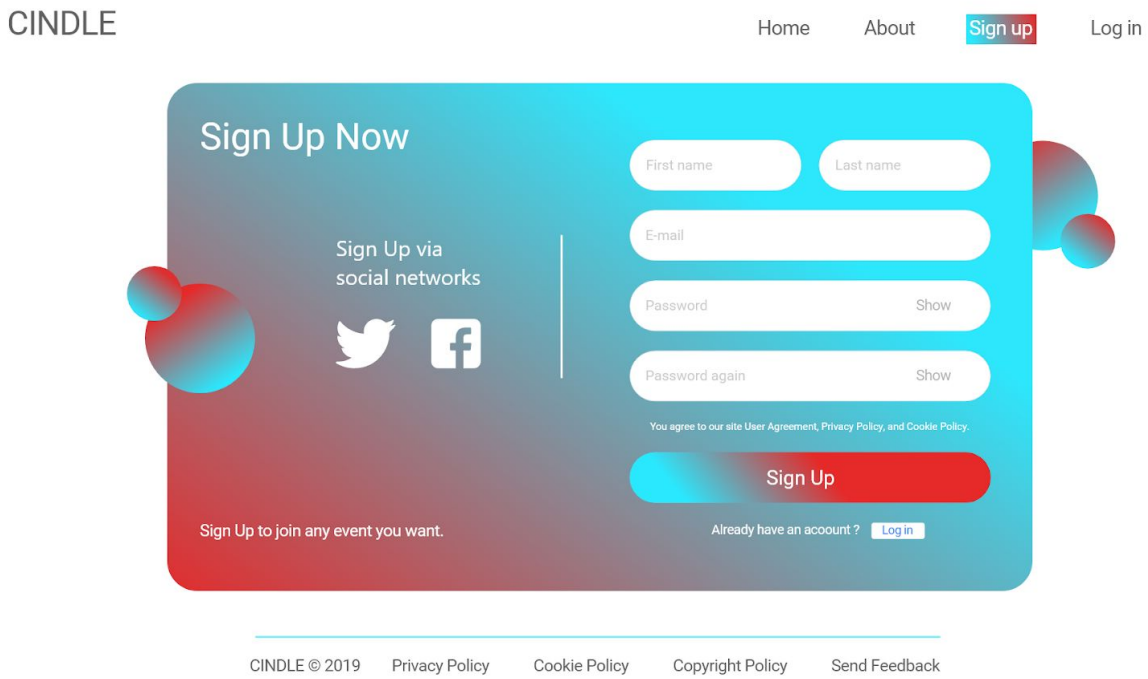
4. Cindle's Forget password Screen:



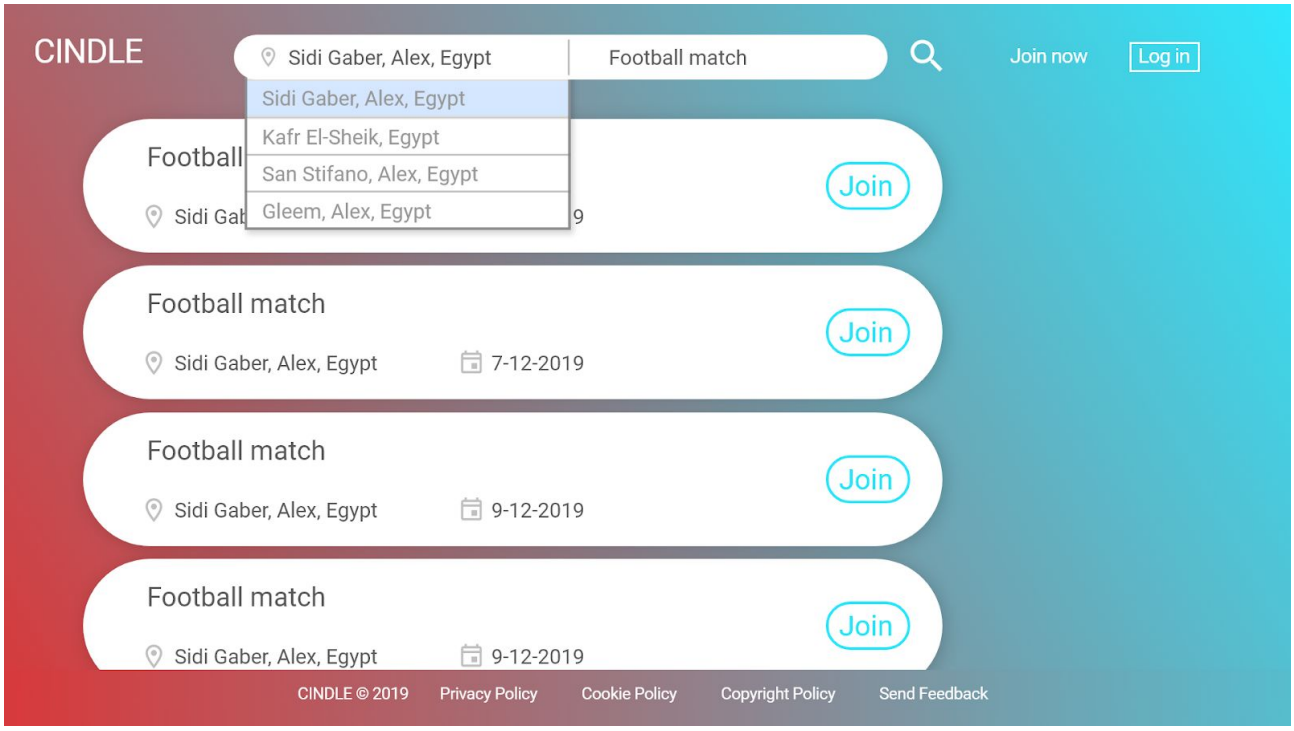
5. Cindle's Forget password result Screen:



6. Cindle's Sign up Screen:

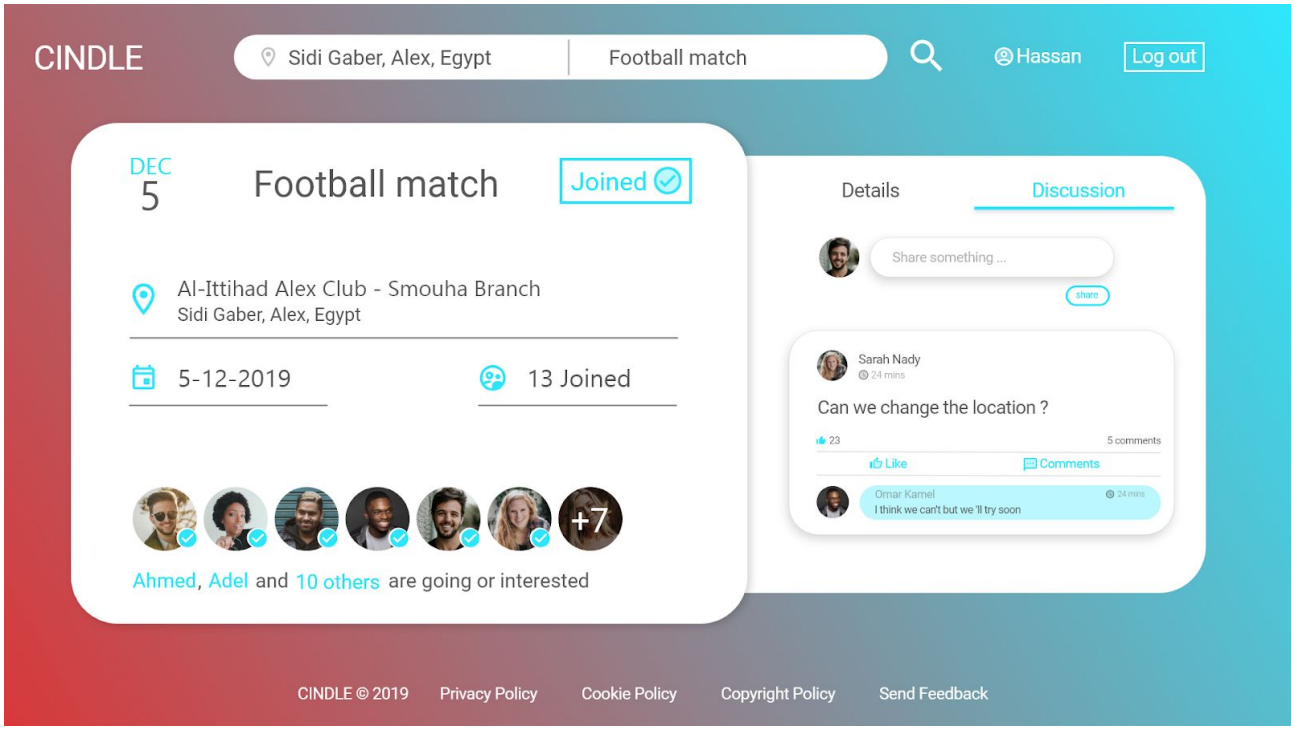


7. Cindle's search result Screen:

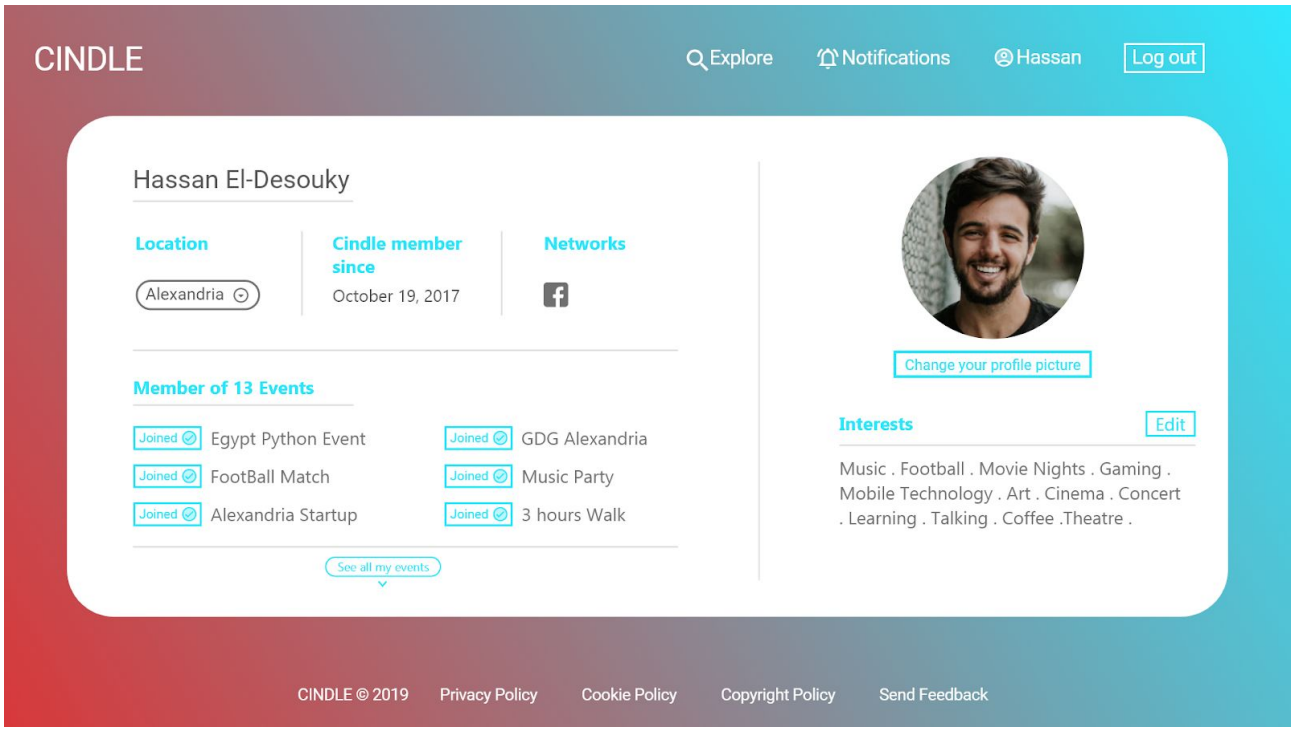


8. Cindle's About:

9. Cindle's Event page:



10. Cindle's Profile page:



3.2 Hardware Interfaces

Since Cindle must run over the internet on Google Chrome that will run on computers equipped with a Pentium 4 processor or higher, which encompasses most machines manufactured since 2001. The computer must have approximately 100MB of free hard drive space and 128MB of RAM. The oldest version of Windows supported by Chrome is Windows XP with Service Pack 2 installed. Chrome also runs on computers with Windows Vista or Windows 7,8 or 10 installed.Also we 'll need web server & database too.

3.3 Software Interfaces

Cindle needs a web browser that supports scripts like Google Chrome or Mozilla Firefox by which users can access the system.

3.4 Communications Interfaces

As Cindle is a social networking website it requires a high-speed internet modem for the use of this application. For the suitable use, there must be a correct internet connection among the users. The system shall use the HTTP protocol for communication over the internet and the internet communication will be through TCP/IP protocol suite.Since Cindle must run over the internet, all the hardware shall require to connect the internet will be a hardware interface for the system. For e.g. Modem, WAN – LAN, Ethernet Cross-Cable.

4. System Features

4.1 Sign-up A Fresh Account

4.1.1 Description and Priority

entering personal information to create an account.
Priority is High.

4.1.2 Stimulus/Response Sequences

The user enters his personal information"name,age,email,phone-number,location,password" and then verifying his account by email.

4.1.3 Functional Requirements

REQ-1: *User must be over 13 years old.*

4.2 Sign-In To An Existing Account

4.2.1 Description and Priority

User entering him/her account information to can join events.
Priority is High.

4.2.2 Stimulus/Response Sequences

entering email/phone number and password.

4.2.3 Functional Requirements

REQ-1: *User must already have an account.*

4.3 Searching Events

4.2.1 Description and Priority

At first, the user searches for the activity which matters to him/her to know more about it's schedule.
Priority is High.

4.2.2 Stimulus/Response Sequences

Searching event list with his filters or keywords or category specified.

4.2.3 Functional Requirements

REQ-1: *Search for already registered activity at our website.*

4.4 Join An Event

4.2.1 Description and Priority

Selecting a specific event to join and view more info about it.
Priority is Medium.

4.2.2 Stimulus/Response Sequences

Choosing the event with matching interest and time, the system let him view the participators of the meetup and the number matched and the left ones .. cutting of the words it will lead the user to the event page including all information inside of it.

4.2.3 Functional Requirements

REQ-1: *Already logged-in in the system.*

4.5 Filling profile with interests

4.1.1 Description and Priority

It's where the user navigates between available interests/activities to look for which is suitable for him/her to share and fill the profile with the selections.
Without these actions, the application would not work properly or it would be useless to use; Priority is High.

4.1.2 Stimulus/Response Sequences

The first step is selecting suitable interests and after proceeding it goes to our database, the user profile is updated and the available event list is also updated.

4.1.3 Functional Requirements

REQ-1: *Verified account, with real personal information not randomly collected.*
REQ-2: *The activity user is searching for is already placed before.*

4.6 Rating Event

4.3.1 Description and Priority

The way how the user ends his little trip with us and let us know how it was.
Priority is Low.

4.3.2 Stimulus/Response Sequences

It pops up with empty five-stars waiting for his response to fill it as much as he liked the total event then select the point he liked/disliked and there's an option to leave a comment specific or general about the event or the point he had chosen.
After submitting the system ask him if he wants to repeat the process" event" in the coming day and wait for the user response with "Yes of course!" or "No thanks".

4.3.3 Functional Requirements

REQ-1: *Correct suggestion list/points in the feedback window.*

4.7 Log out & End Session

4.2.1 Description and Priority

Ending the session on the website and sign out.

Priority is low.

4.2.2 Stimulus/Response Sequences

Every header includes a Log-out button which proceed the action.

4.2.3 Functional Requirements

REQ-1: *Already logged-in in the system.*

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The application must contain a limit for interests that the user can select, that is because if a user selects too many interests, this may affect the performance of his or her meet list. The application must support approximately 10000 users at the same time based on the population of the city.

5.2 Safety Requirements

The application mustn't allow anyone to join it unless he or she is over 13 years old. It should also allow the user to know who else is going to the meeting.

5.3 Security Requirements

User identity must be validated before being allowed to use the site. The quality and safety of the place must be verified before doing any meeting. Make sure the user agrees to all privacy & policy of the app.

5.4 Software Quality Attributes

The application will work 24/7 if an internet connection is available. The server will be maintainable. We will do a digital test and on-site test to make sure that the app is working properly. The application will contain frequently asked questions "FAQ" to help the user in an easy way. There will be continuous maintenance of the application. The user can open the application from any device (Android, iOS, browser). The user can use the application from anywhere in the world.

5.5 Business Rules

1. Profit from places for more meetings "long-term plan".
2. Profit through Advertisements.

6. Other Requirements

As Google Places and Google Maps API is being used for this application, it is mandatory that we abide by the terms of use specified by Google, database storage and domain with host, internationalization needed for multiple languages users, the places in which event will be held in.

Appendix A: Glossary

No glossary terms available at this time.

Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

Appendix C: Use Case Diagrams

The Use Case Diagrams will be attached in a seperate file.

Appendix D: UI(user interface)

The user interface (UI) is the point of human-computer interaction and communication in a device. This can include display screens, keyboards, mouse and the appearance of a desktop.