CELEBREMOS

Software Requirements Specification

Version 3.0

06/30/2019

Hari Priya Jupally

Karun Bourishetty

Indra Reddy Mamidi

Vishal Reddy Pannala

Saikumar Nalivela

Poojitha Singam

Sairam Badisa

Submitted in partial fulfillment

Of the requirements of

CSIS 44-691 Graduate Directed Project 1

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Description** | **Author** | **Comments** |
| 06/11/2019 | Version 1.0 | Poojitha Singam | The initial gathering of requirements by understanding the evite website. |
| 06/12/2019 | Version 2.0 | Hari Priya Jupally | Updated requirements and ER diagram. |
| 06/13/2019 | Version 3.0 | Indra Reddy | Created prototype |
| 06/30/2019 | Version 4.0 | Sai Ram Badisa, Poojitha Singam,  Hari Priya Jupally | Technical Manual |

# **Document Approval**

The following Software Requirements Specification has been accepted and approved by the following:

|  |  |  |  |
| --- | --- | --- | --- |
| **Signature** | **Printed Name** | **Title** | **Date** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

**Table of Contents Page Number**

1.Introduction

1.1.Purpose

1.2.Scope

1.3.Definitions, Acronyms, and Abbreviations

1.4.References

1.5.Overview

2.General Description

2.1.Product Perspective

2.2.Product Functions

2.3.User Characteristics

2.4.General Constraints

2.5.Assumptions and Dependencies

3.Specific Requirements

3.1.External Interface Requirements

3.1.1.User Interfaces

3.1.2.Hardware Interfaces

3.1.3.Software Interfaces

3.1.4.Communications Interface

3.2. Functional Requirements

3.3. Use Cases

3.4. Class/Objects

3.5. Non-Functional Requirements

3.5.1. Performance

3.5.2. Reliability

3.5.3. Availability

3.5.4. Security

3.5.5. Portability

3.6. Inverse Requirements

3.7. Design Constraints

3.8. Logical Database Requirements

3.9. Other Requirements

3.10. Prototypes (for complete project)

3.11. Use Case Diagrams

4. Design

4.1. ER diagram

4.2. GUI

5.Analysis Models

5.1. Data Flow Diagram

5.2. Sequence Diagram

## 

**1.** **Introduction**

**1.1 Purpose**

The main purpose of the application is to create, modify and delete an online invitation system which helps the people to invite guests for their party to manage the number of people attending the event, creating polls to finalize the food menu and what costume they have to wear. This application helps the event organizers to plan the event successfully without any issues.

**1.2 Scope**

The scope of the project is to create an application that helps the event organizers to invite the guests to party which helps the organizers to plan for the place, food items, number of guests, what are the lists of items they need to bring while coming to the party and what costume they have to wear.

**1.3 Definitions, Acronyms, and Abbreviations**

RSVP - Répondez s'il vous plaît (Please respond)

ER - Entity Relationship

**1.4.References**

[**https://www.evite.com/event/create?template=general\_dyo&event\_type=&c=featured&ctrk=featured&op\_exp=quick\_create&expand\_gallery**](https://www.evite.com/event/create?template=general_dyo&event_type=&c=featured&ctrk=featured&op_exp=quick_create&expand_gallery)

**1.5.Overview**

**2.General Description**

**2.1.Product Perspective**

This application is making the user to create or select any template or invitation and then after selecting the invitation, the user is going to send the invitation to all the guests through mail or text to the phone. The user can choose among the different invitations and send them using this application.

**2.2.Product Functions**

This application does not have any admins. It consists of only users. Users can create an invitation and invite the guests.

**User can perform all the below operations:**

* Sign up and log in to the application.
* If the user forgets the password, then the user can reset the password.
* Can choose the type of invitation.
* Ability to create the groups so that next time they can be invited very easily.
* Responses from the guests can be taken as the preferred food, no.of guests coming, etc.
* Users can create a poll and take the responses from the guests.

**Guests can perform the below Operations:**

* Able to login to the application.
* Can view all the invitations they’ve been invited to.
* Answer to the polls for which the hosts have created.
* Manage all the invitations.
* Giving the RSVP

**2.3.User Characteristics**

This application can be used by any users to create any invitation and invite to any event.

**2.4.General Constraints**

While using the servers there might be a chance of application not working properly.

**2.5.Assumptions and Dependencies**

The people who are working in this project have a clear idea of the requirements.

**3.Specific Requirements**

**3.1.External Interface Requirements**

**3.1.1.User Interfaces**

The UI is designed and developed by using Angular.

**3.1.2.Hardware Interfaces**

**3.1.3.Software Interfaces**

* Angular to develop the front end.
* MongoDB as a database.
* Express as a framework between frontend and backend.
* Node js is used to run the application on the server.

**3.1.4.Communications Interface**

1. Google Docs
2. Whats App
3. Gmail

**3.2. Functional Requirements**

1. Create a Homepage
2. Create the Login page  
    A. Create the register page  
    B. Create a sign-in page  
    C. Managing forgot password.
3. Creating profile page
4. Creating the events page.
5. Create an invitation template  
    A. Create templates for birthday.  
    B Create templates for marriage.  
    C. Create templates for graduation.
6. Create details page of template  
    A. Modifying the details of template  
    B. Customizing the RSVP style (Please respond)
7. Creating the guest list for sending the invitation.  
    A. Send mail through mail-id  
    B. Send text message through phone number
8. Managing the event page  
    A. Create the details tab  
    B. Create messaging tab  
    C. Create managing guest tab  
    D. Creating what to bring tab  
    E. Creating polls  
    F. Setting tab

**3.3. Use Cases**

**3.4. Class/Objects**

**3.5. Non-Functional Requirements**

**3.5.1. Performance**

**3.5.2. Reliability**

**3.5.3. Availability**

**3.5.4. Security**

**3.5.5. Portability**

**3.6. Inverse Requirements**

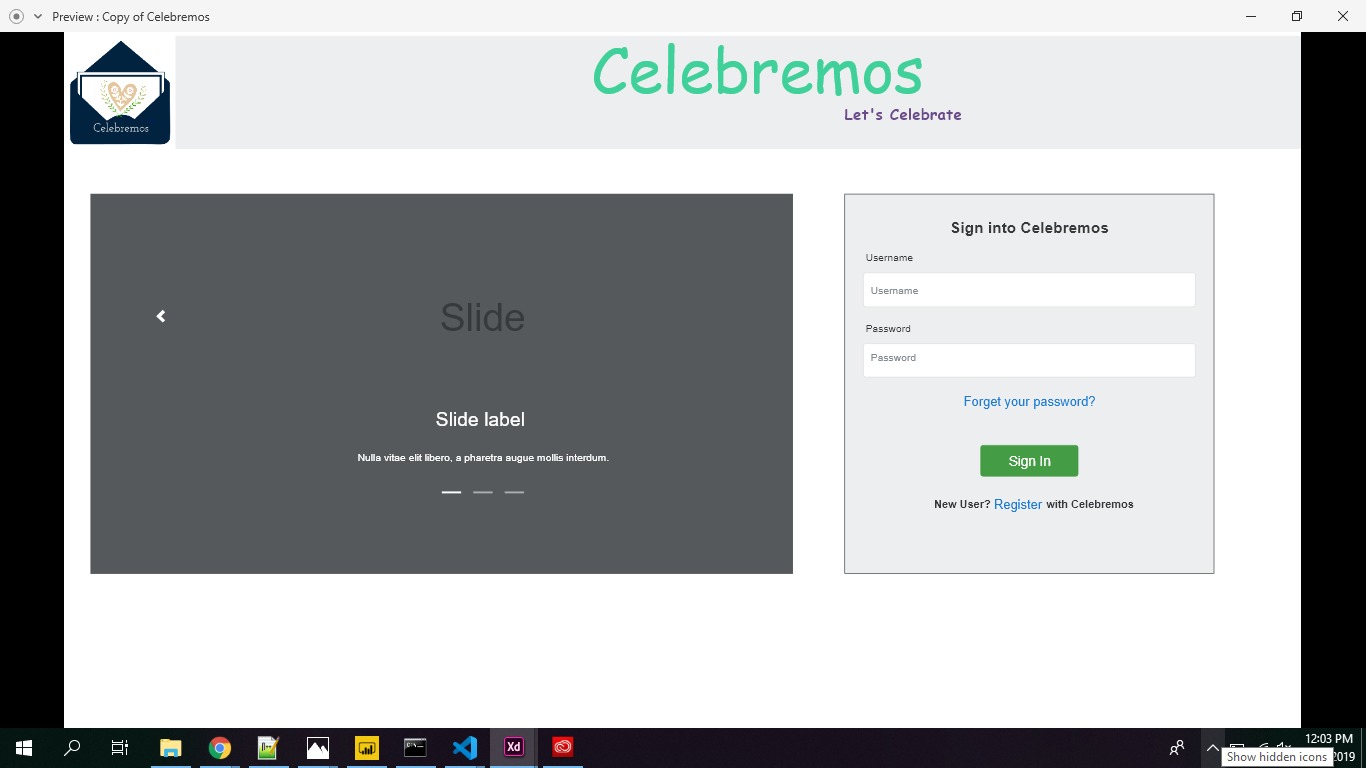
**3.7. Design Constraints**

**3.8. Logical Database Requirements**

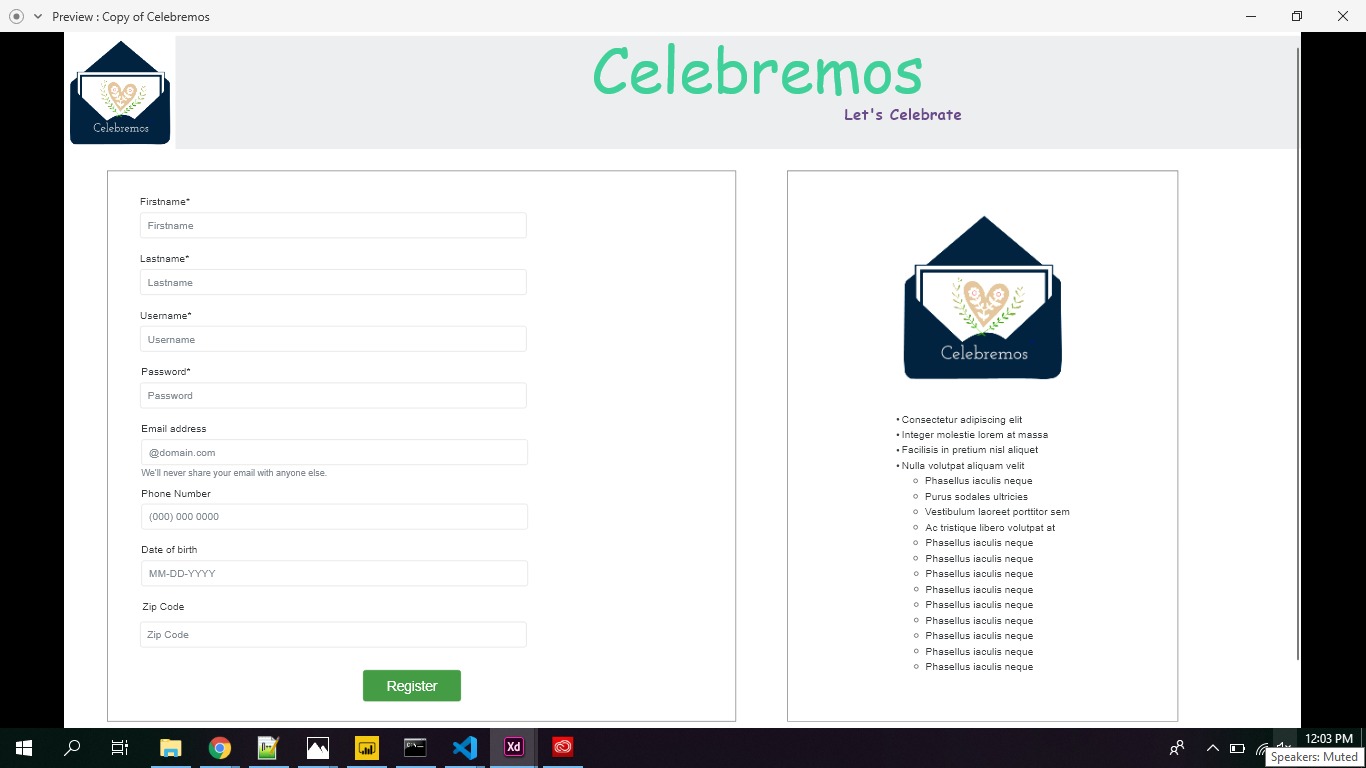
**3.9. Other Requirements**

**3.10. Prototypes (for complete project)**

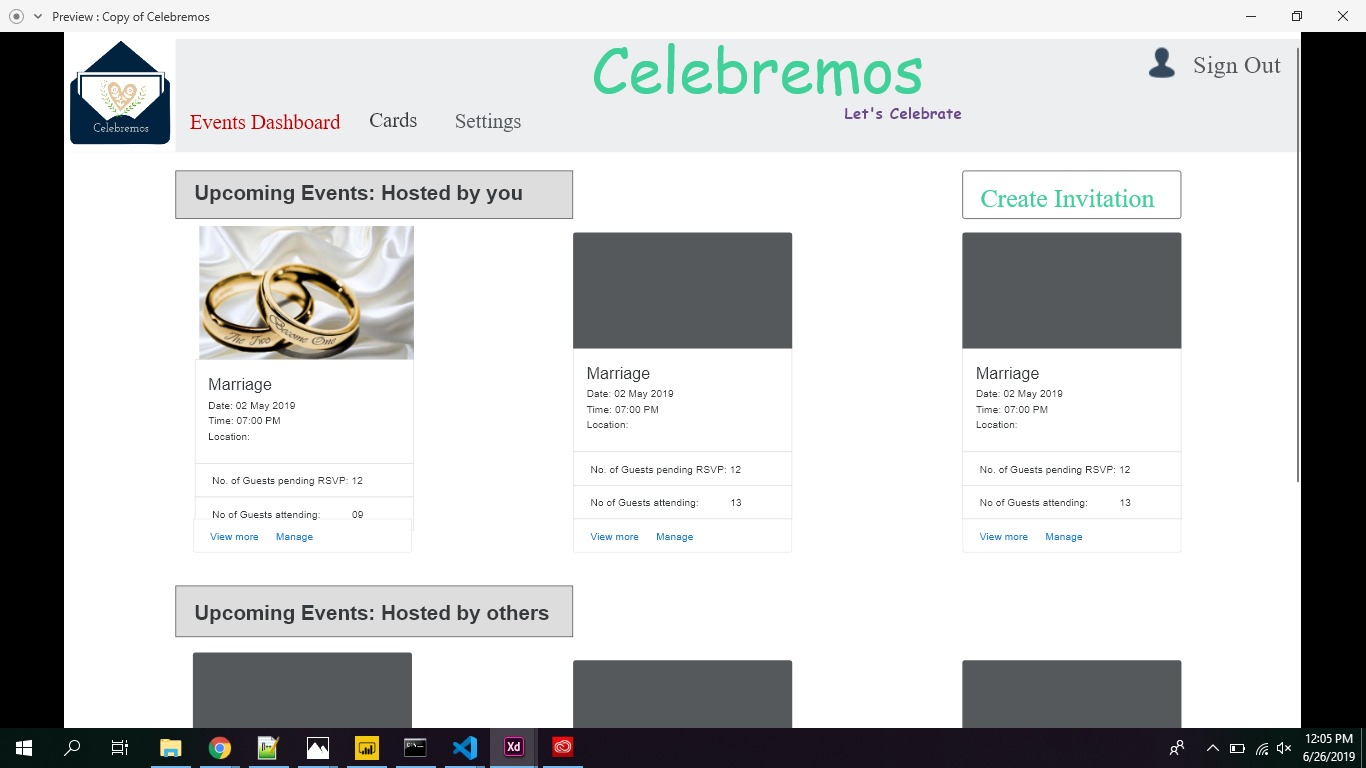
**Login Page**

****

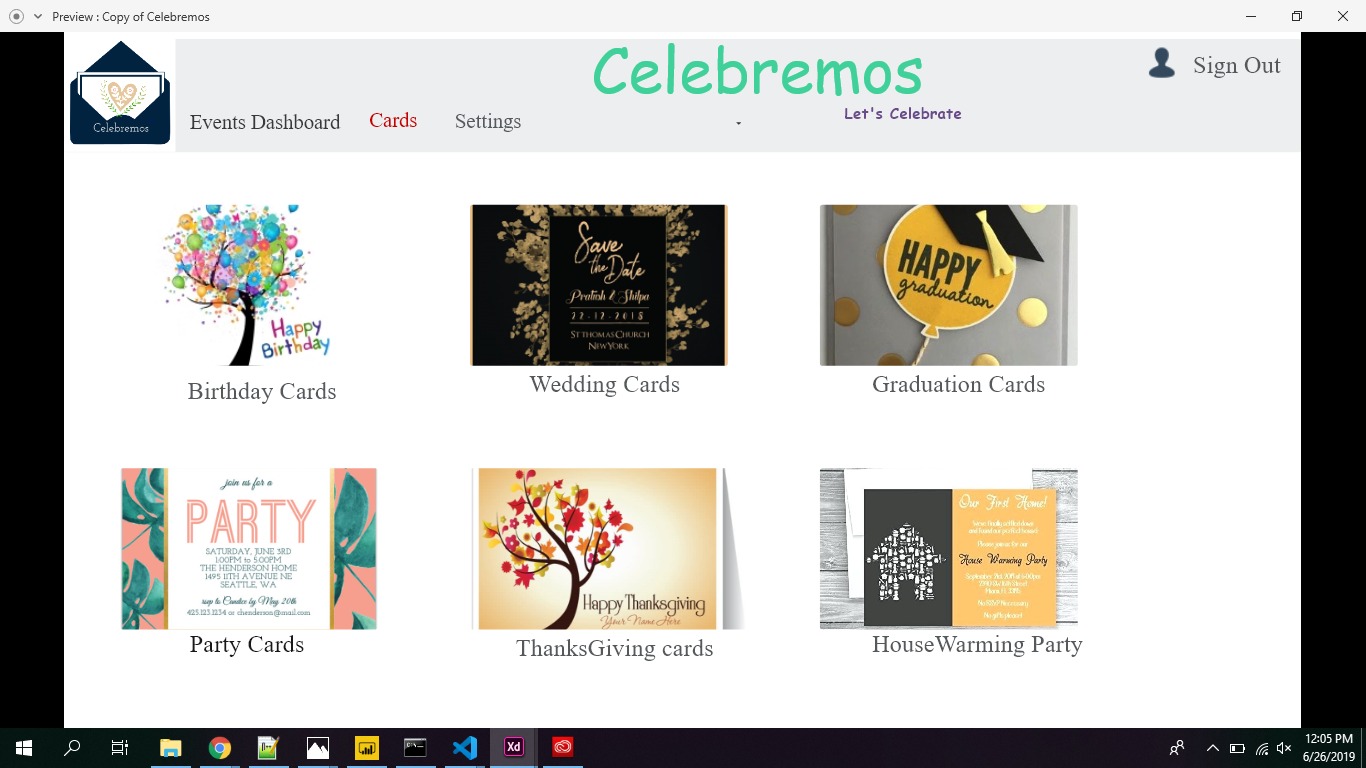
**Registration**

****

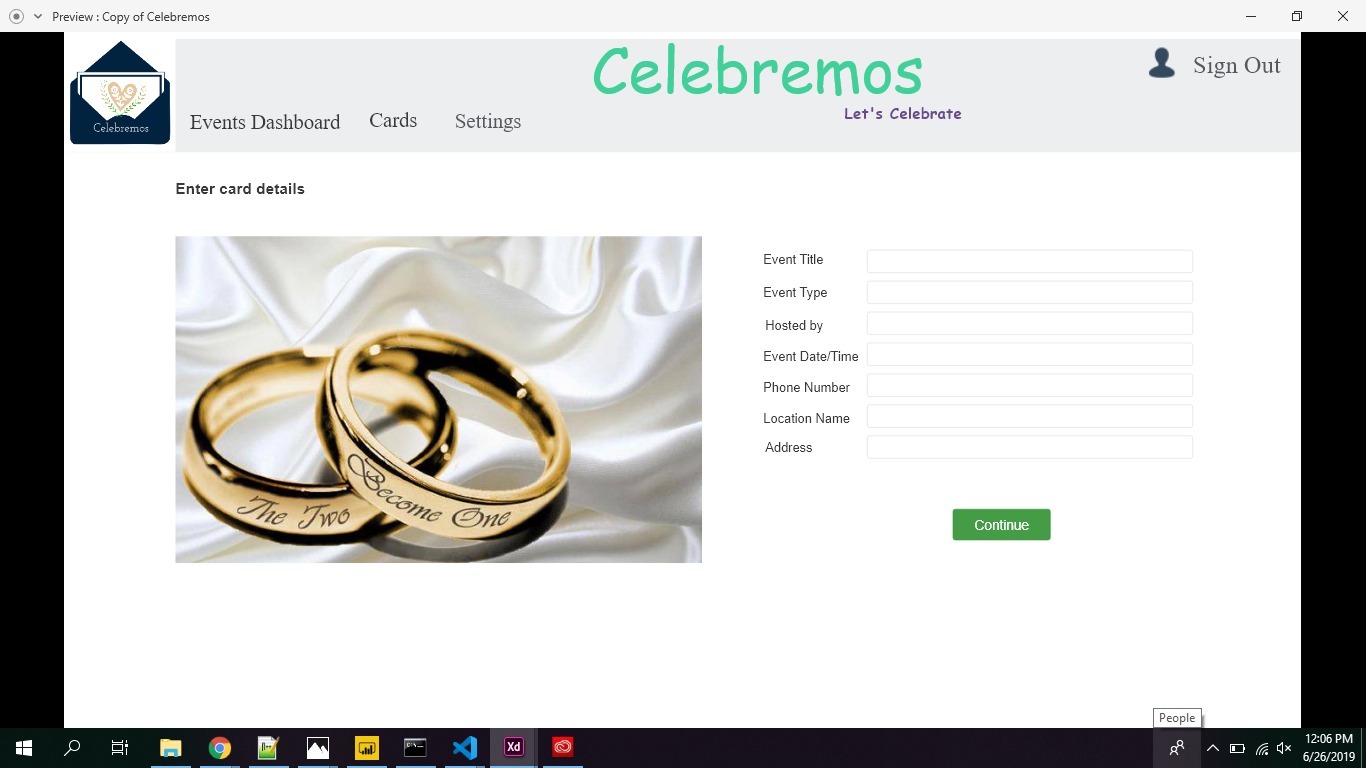
**Event Dashboard**

****

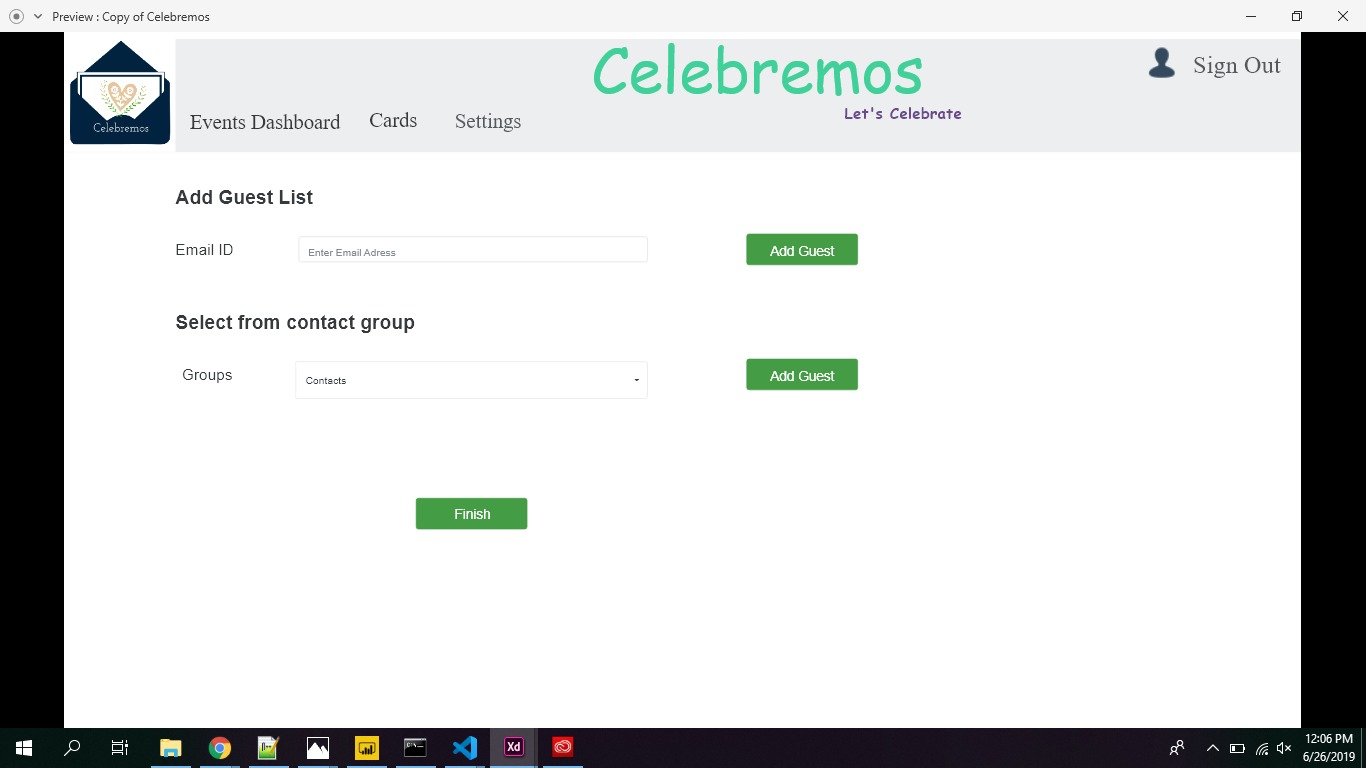
**Create Invitation**

****

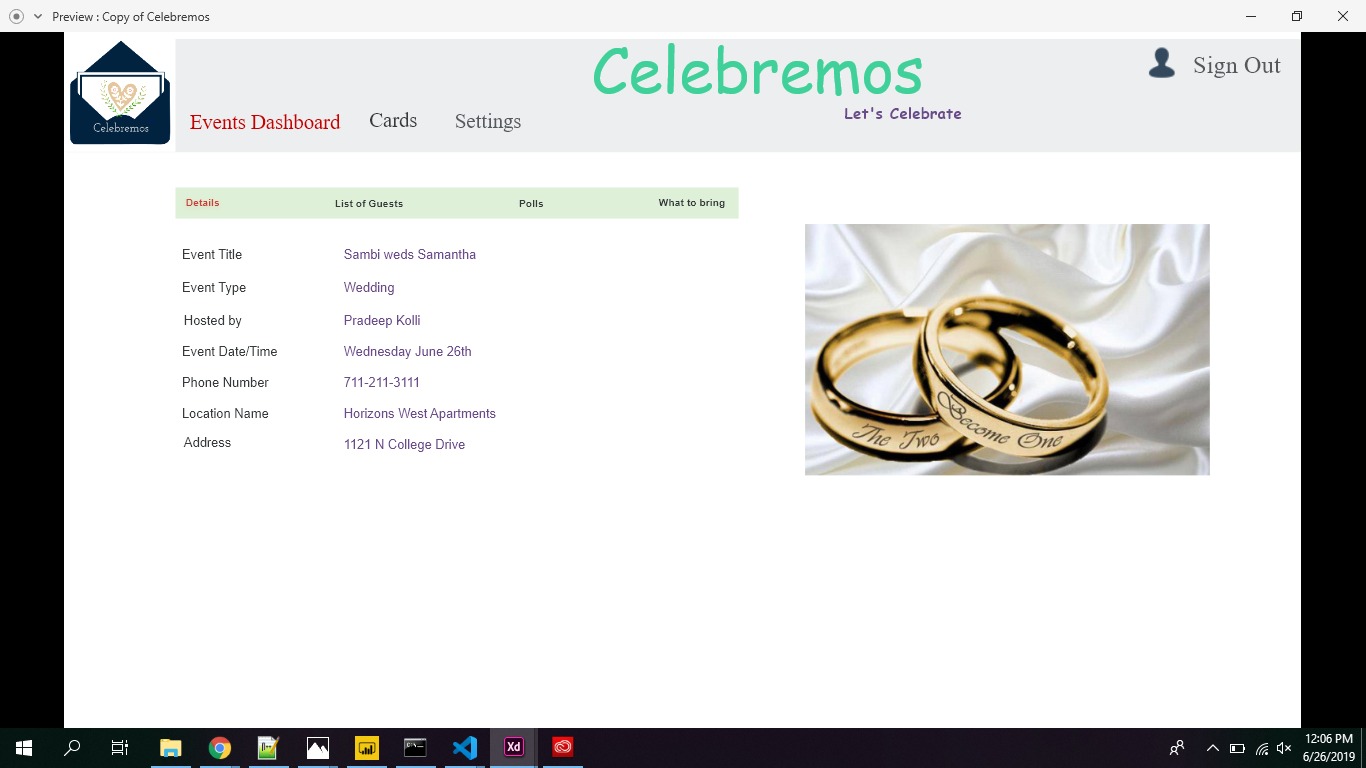
**Card Details Page**

****

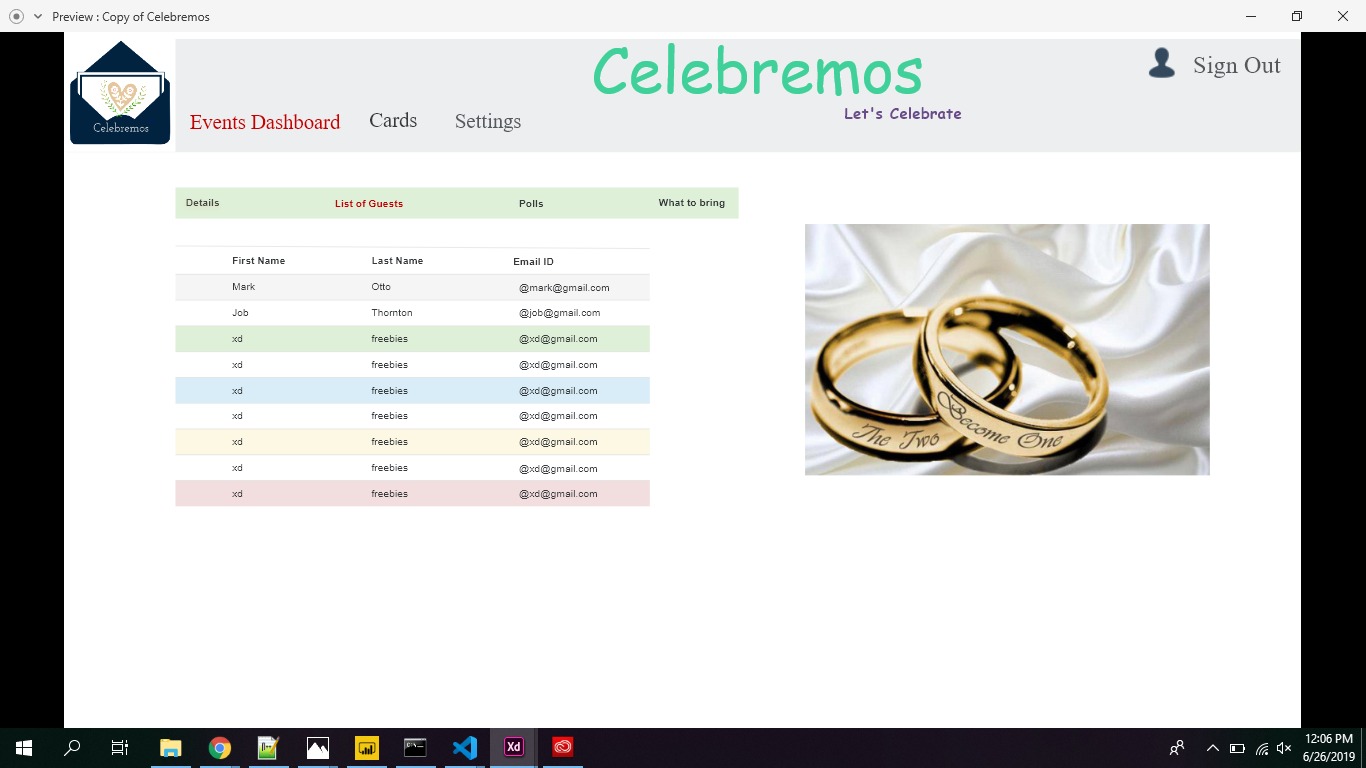
**Adding Guests**

****

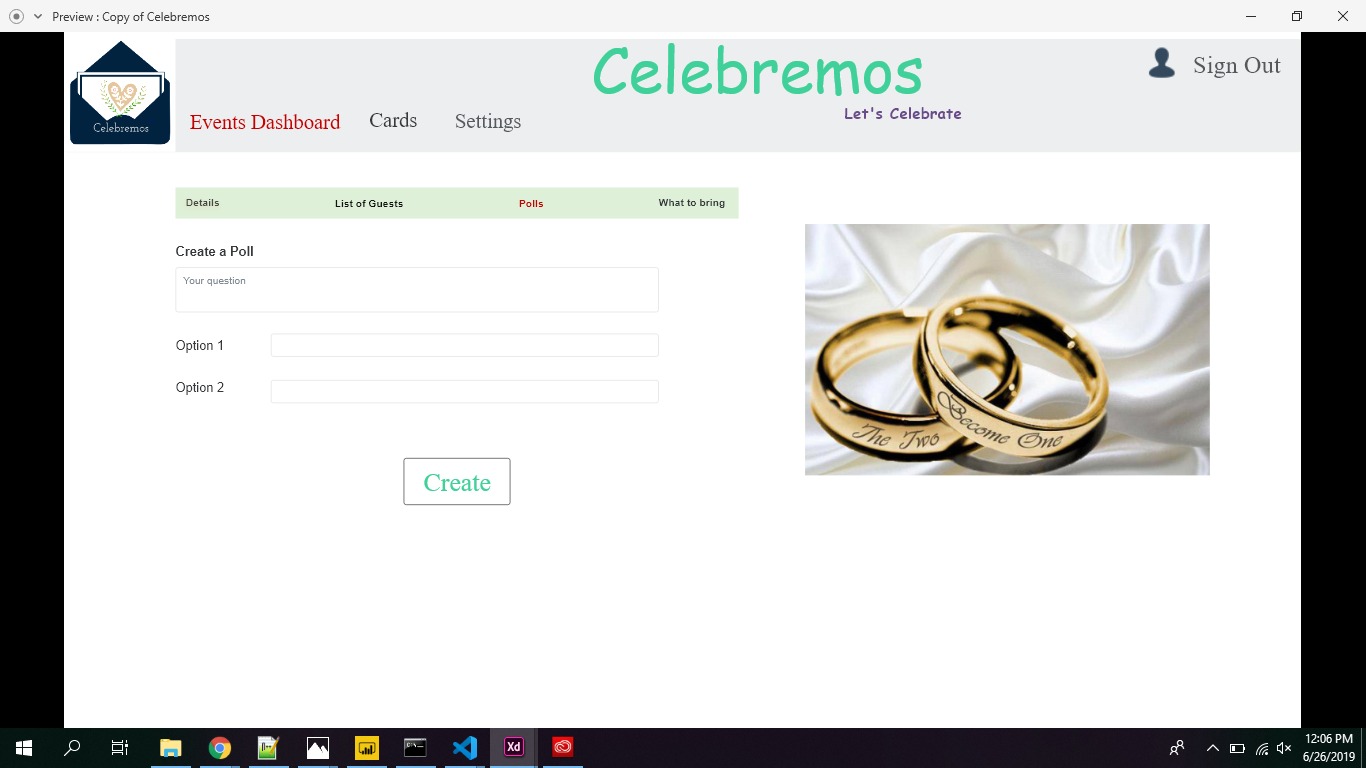
**Event Details Page**

****

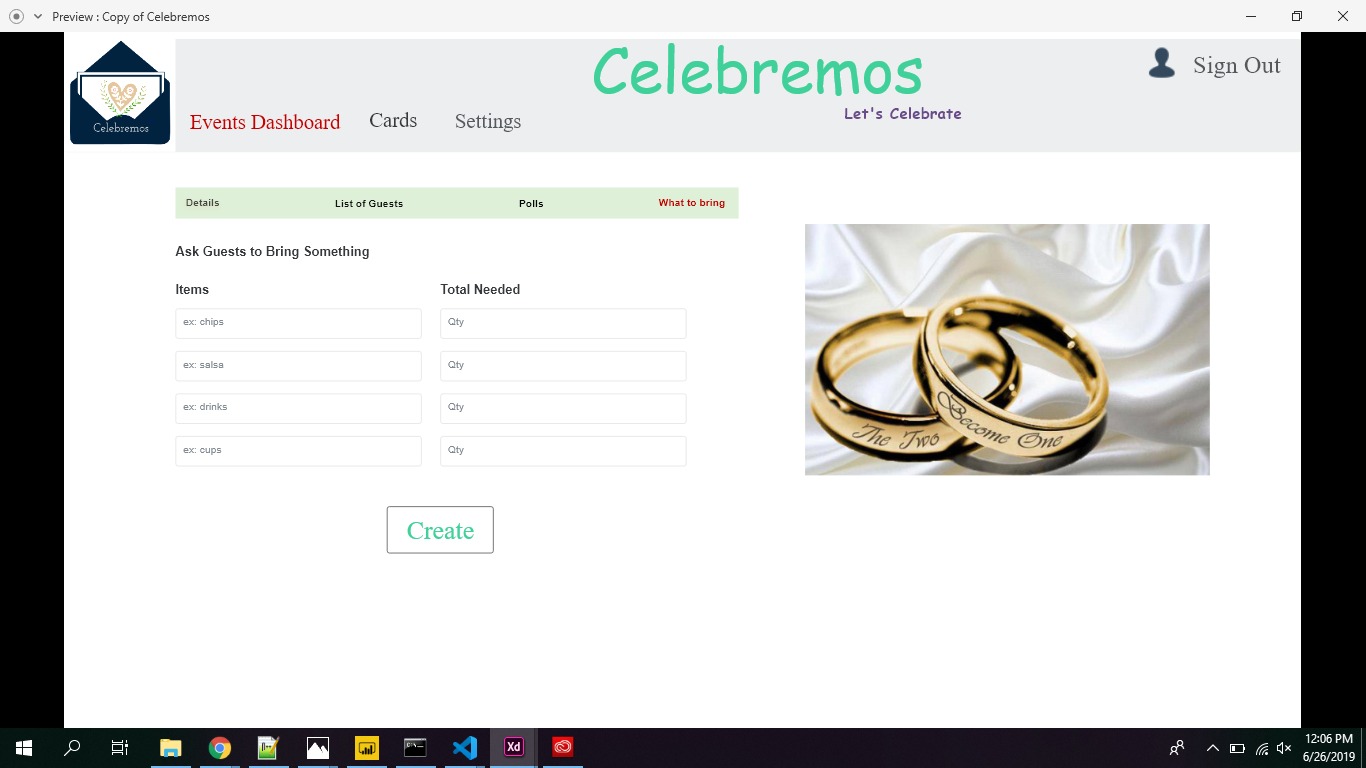
**List of Guests Page**

****

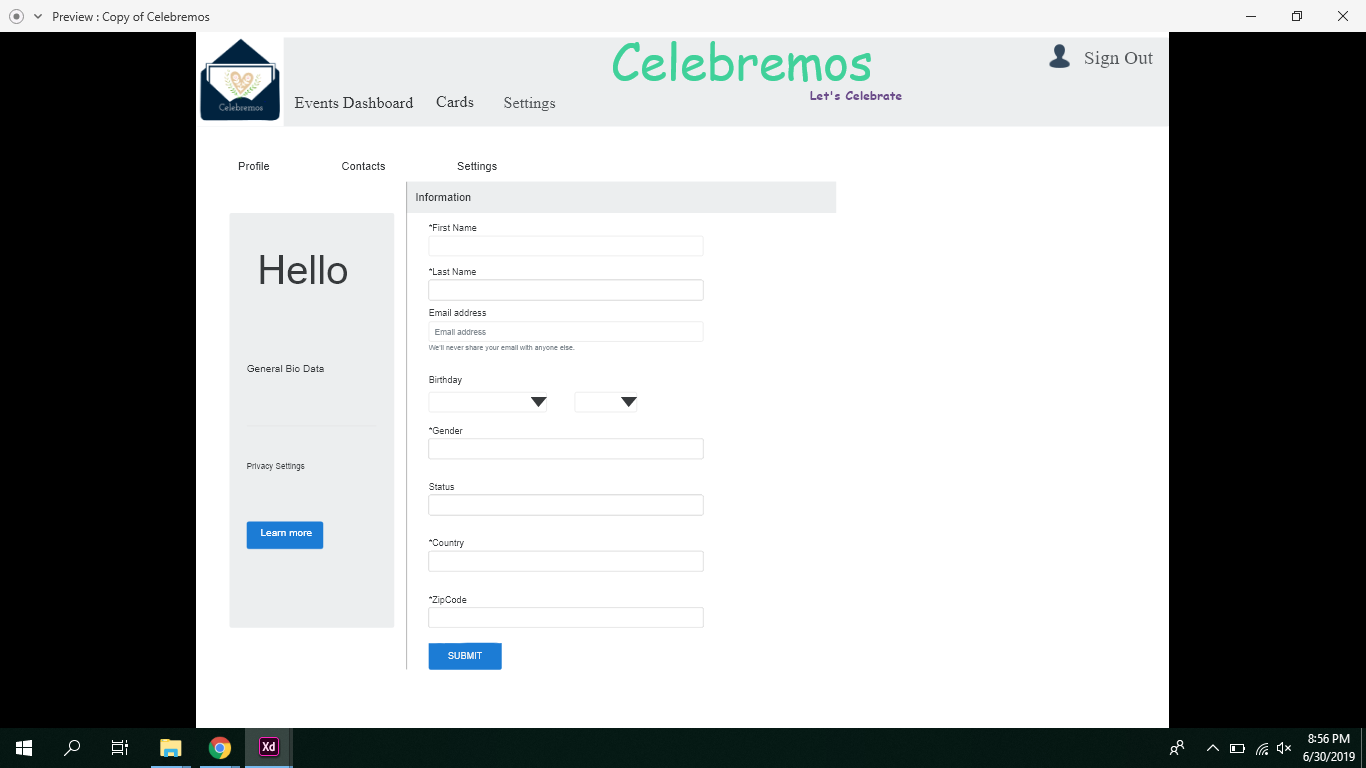
**Creating Polls**

****

**What to bring page**

****

**Profile Page**

****

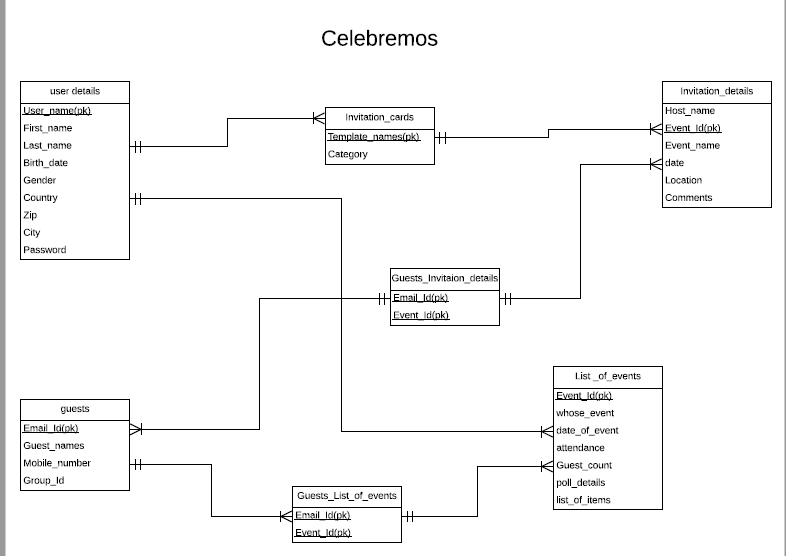
**Preview Link:**

<https://xd.adobe.com/view/0ea15000-5391-4675-56b6-b7d9a39e6ce7-7890/>

**3.11. Use Case Diagrams**

**4. Design**

**4.1. ER diagram**

****

The above ER diagram has 6 entities and 1 associative entity. A user can see some Invitation cards and use an invitation card to send an invitation. User can add any number of guests and see all the events he has hosted or invited to. He can manage events and RSVPs.

**4.2. GUI**

**5.Analysis Models**

**5.1. Data Flow Diagram**

**5.2. Sequence Diagram**

# **6. Technical Manual:**

## **6.1. Document Identification:**

This part of the documentation list each component of the application with code description and the snapshot of it. Since there is very little time in this semester we’ve worked mostly on the login functionality.

* User Signup.
* User login.
* Creating an invitation.
* Choosing or creating a template.
* Creating the polls and taking the inputs from guests.
* View the invitations.

## **6.2. System overview:**

We can create a couple of invitations like

* Birthdays
* Marriages
* Graduation

The application is designed in such a way that the user can choose or create an invitation by the users based on the requirements. All the required events requirements are included in this scope of the document.

## **6.3. Tools and technologies**

### **6.3.1. MongoDB**

MongoDB stores data in flexible, JSON-like documents, meaning fields can vary from document to document and data structure can be changed over time.

### **6.3.2 Express**

Express is used as a web application framework for Node. It handles the interactions between the front-end and the database.

**6.3.3 Angular**

AngularJS is a structural framework for dynamic web apps. It lets you use HTML as your template language and lets you extend HTML's syntax to express your application's components clearly and succinctly.