

# CelebrateMate: Automated Birthday and Anniversary Reminders

## Project Report

### Table of Contents

1. Introduction
2. Market Research
3. Define Requirements
4. Design and Prototyping
5. Development
6. Integration
7. Testing
8. Deployment
9. Maintenance and Support
10. Conclusion

### 1. Introduction

CelebrateMate is an automated event reminder application designed to help users keep track of important dates such as birthdays and anniversaries. This project aims to deliver a reliable and user-friendly solution that integrates with various communication channels to send reminders. The following sections outline the steps taken for the successful implementation of CelebrateMate.

## 2. Market Research

Market research was conducted to understand the current landscape of event reminder applications. Key findings included:

- Existing Solutions: Several event reminder apps are available, but many lack comprehensive integration with SMS and email services.
- Target Demographics: The primary users are busy professionals, forgetful individuals, and people with extensive social networks.
- Competitive Landscape: Competing applications often have limited customization options and poor user interfaces.

Based on this research, CelebrateMate aims to address these gaps by offering robust integration, customization, and a user-friendly interface.

## 3. Define Requirements

Collaboration with stakeholders helped define the functional and non-functional requirements:

-Functional Requirements:

- User authentication and profile management.
- Event creation and management (birthdays, anniversaries).

- Reminder notifications via SMS and email.
- Calendar synchronization and contact management.
- Non-Functional Requirements:
  - Scalability to handle growing user base.
  - High reliability and uptime.
  - Security measures to protect user data.
  - User-friendly interface for all age groups.

## 4. Design and Prototyping

### Wireframes and Prototypes

- Wireframes: Initial sketches were created to visualize the layout of the application.
- Prototypes: Interactive prototypes were developed using tools like Figma to simulate user interactions and gather feedback.
- Usability Testing: Conducted with a small group of users to identify and address any usability issues before development.

## 5. Development

### Backend Development

- Technology Stack: Python with Flask framework, MongoDB for database management.
- Infrastructure: Hosted on cloud services for scalability and reliability.
- APIs: Developed RESTful APIs for user management, event handling, and notification services.

### Frontend Development

- Technology Stack: HTML, CSS, JavaScript.
- Components: Developed reusable components to ensure consistency and efficiency.

## 6. Integration

### Third-Party APIs

- SMS and Email Delivery: Integrated with Twilio for SMS and SendGrid for email services.
- Calendar Synchronization: Integrated with Google Calendar API.
- Contact Management: Integrated with various social media and email contact APIs.

## 7. Testing

## Comprehensive Testing

- Unit Testing: Ensured individual components function correctly.
- Integration Testing: Verified that integrated components interact as expected.
- User Acceptance Testing: Gathered feedback from a diverse group of users to ensure the application meets user expectations.

## Cross-Platform Compatibility

- Tested on various devices and browsers to ensure consistent performance and functionality.

## 8. Deployment

### Production Environment

- Hosting: Deployed on AWS for robust performance and scalability.
- Monitoring: Implemented monitoring tools to track performance metrics and identify issues.
- Initial Rollout: Managed a phased rollout to address any initial issues promptly.

## 9. Maintenance and Support

### Ongoing Maintenance

- Regular updates to fix bugs and improve performance.
- Continuous integration and delivery pipeline to streamline updates.

### User Support

- Dedicated support team to address user feedback and issues.
- Feature enhancements based on user suggestions to keep the application relevant.

## 10. Conclusion

CelebrateMate has been developed through a systematic and thorough process, from market research to deployment. The focus on user needs and robust integration ensures that CelebrateMate stands out in the competitive landscape of event reminder applications. Ongoing maintenance and user support will help maintain its relevance and success in the market.

This report provides a comprehensive overview of the CelebrateMate project, highlighting the critical steps and methodologies used to deliver a high-quality event reminder application.