**Software Requirements Specification (SRS)**

**Project Title:** Event Planner  
**Presented by:** Dalia Gamal  
**Date:** April 2025

**1. Introduction**

**1.1 Purpose**

This document describes the functional and non-functional requirements of the **Event Planner** system. The system is designed to help users manage events, guests, and invitations in an organized and efficient way.

**1.2 Scope**

The system allows users to:

* Add, update, and delete **guests**.
* Add, update, and delete **events**.
* Send and manage **invitations** with RSVP status.
* View all data in an organized UI using **DataGridViews**.
* Use a central **Main Form** to navigate between modules.
* Automatically update views after any action.
* **Search** for data inside each module (Guests, Events, Invitations).

**1.3 Intended Audience**

* Developers.
* Supervisors or instructors.
* End-users interested in using the system.

**1.4 System Overview**

The system is built using **C# Windows Forms** and **Entity Framework (Code First)**. It is designed for desktop use with a clean and professional UI.

**2. Overall Description**

**2.1 Product Perspective**

This is a stand-alone desktop application built using Windows Forms. It connects to a local database using Entity Framework.

**2.2 Product Functions**

* Manage event records (create/edit/delete).
* Manage guest records (create/edit/delete).
* Manage invitations and their RSVP status.
* Display data in real-time.
* Search in data tables.
* Switch between modules via a main navigation form.

**2.3 User Classes and Characteristics**

* **Admin/User:** Can access all functionalities.
* No login or role-based restrictions implemented (optional for future updates).

**2.4 Operating Environment**

* .NET Framework 4.7.2+
* Visual Studio 2022+
* Windows 10/11

**2.5 Constraints**

* System runs locally, no cloud or multi-user support.
* Entity Framework Code First used, so initial database is auto-generated.

**3. Functional Requirements**

| **ID** | **Feature Description** |
| --- | --- |
| FR1 | User can add/edit/delete **Guests** with name, email, phone. |
| FR2 | User can add/edit/delete **Events** with title, location, date. |
| FR3 | User can send **Invitations** and select RSVP status (Pending, Confirmed, Declined). |
| FR4 | A **Main Form** allows navigation between Guests, Events, and Invitations. |
| FR5 | **Data updates automatically** after each action (Add/Edit/Delete) without manual refresh. |
| FR6 | Users can **Search** records in Guests, Events, and Invitations based on fields (e.g., name, title). |
| FR7 | UI is styled with a **dark theme**, readable fonts, and consistent layout. |

**4. Non-Functional Requirements**

| **ID** | **Requirement** |
| --- | --- |
| NFR1 | The system should be responsive and not lag during CRUD operations. |
| NFR2 | The UI should be intuitive and user-friendly. |
| NFR3 | Styling should follow a dark theme: Black background, DimGray controls, WhiteSmoke text. |
| NFR4 | All features should be accessible with minimal clicks. |
| NFR5 | System should handle invalid inputs and show appropriate messages. |

**5. System Models (Optional)**

**5.1 Entity Relationships**

* Guest ↔ Invitation ↔ Event  
  One guest can have many invitations.  
  One event can have many invitations.

**5.2 UI Mockups (Simplified)**

* Main Form with buttons: [Guests] [Events] [Invitations]
* Each module shows:
  + Input fields
  + DataGridView
  + Buttons: Add / Edit / Delete / Search /About

**Conclusion**

This document outlines the core functionality of the Event Planner system. All requirements mentioned are implemented or in-progress, including the 7 features needed for the project to be considered complete.