SRS

WHATSAPP CLONE

Table of Contents

1. Introduction

CareerPath is a web Application where student Provide date and Information and ofter that Career Path will Provide the Best Combination of Courses and collage. which is for suitable for student's Career Path Ultimately, Career Path enpowers students to make informed decisions about their future.

1.1 Purpose

The purpose of this document is to outline the functional requirements for the development of a software solution based on the provided case study. This SRS document aims to serve as a comprehensive guide for the development team to understand the functional aspects of the software.

1.2 Scope

This SRS document covers the functional requirements of the software solution as described in the provided case study. It includes detailed specifications of the software's features, functionalities, and interactions.

1.3 Definitions, Acronyms, and Abbreviations

SRS:	Software Requirement Specification
UI:	User Interface
DB:	Database
API:	Application Programming Interface
GUI:	Graphical User Interface

1.4 *References*

The references for the above software are as follows:-

i.www.google.co.in

ii.www.winkipedia.com

IEEE. Software Requirements Specification Std. 830-1993.

2. General Description

2.1 Product Perspective

The software solution described in this SRS is an application for managing and improving the career path selection process for students. It interacts with users through a user-friendly web interface. It may also integrate with external databases and services for enhanced functionality.

2.2 Product Features

The software shall provide the following key features:

2.2.1 User Registration and Login

- Users can register for an account.
- Users can log in using their credentials.

2.2.2 Career Path Analysis

- The system shall analyze user choices and interests.
- It shall provide advice on the best career options for the user.
- It shall determine suitability for a specific career or role type.

2.2.3 Form Submission

- Users can fill out a form providing their information, including their full name, class, subject, graduation, and interests.
- The system shall process this data for career path analysis.

2.2.4 University Information

- Display information about universities, including logos and details.

2.3 User Classes and Characteristics

The software is primarily designed for two user classes:

- Students: Users who are looking for career guidance.
- Administrators: Users who manage the system's content and user data.

3. Specific Requirements

3.1 User Interface Requirements

3.1.1 Registration and Login

- Users shall be able to register with a valid email address and password.
- Users shall be able to log in with their registered credentials.

3.1.2 Form Page

- Provide a user-friendly form page for inputting personal information.
- Use JavaScript to show the user's name after form submission.

3.1.3 Results Page

- Display analysis results based on user inputs.
- Show recommended courses, entrance exams, and top colleges.

3.2 Functional Requirements

3.2.1 Career Path Analysis

- The system shall analyze user choices and interests to provide career advice.

3.2.2 University Information

- The software shall display university logos and details.

4. External Interfaces

4.1 Database

- The software shall interact with a database to store user information and analysis results.

5. Constraints

- The system should be developed using HTML, CSS, JavaScript for the front-end and a backend technology of choice (e.g., Node.js, Python, Ruby) for server-side processing.
- The system should comply with data protection and privacy regulations.

SRS

Whatsapp Web Clone

Table of Contents

1. Introduction

CareerPath is a web Application where student Provide date and Information and ofter that Career Path will Provide the Best Combination of Courses and collage. which is for suitable for student's Career Path Ultimately, Career Path enpowers students to make informed decisions about their future.

1.1 Purpose

The purpose of this document is to outline the functional requirements for the development of a software solution based on the provided case study. This SRS document aims to serve as a comprehensive guide for the development team to understand the functional aspects of the software.

1.2 Scope

This SRS document covers the functional requirements of the software solution as described in the provided case study. It includes detailed specifications of the software's features, functionalities, and interactions.

1.3 Definitions, Acronyms, and Abbreviations

SRS:	Software Requirement Specification
UI:	User Interface
DB:	Database
API:	Application Programming Interface
GUI:	Graphical User Interface

1.4 *References*

The references for the above software are as follows:-

i.www.google.co.in

ii.www.winkipedia.com

IEEE. Software Requirements Specification Std. 830-1993.

2. General Description

2.1 Product Perspective

The software solution described in this SRS is an application for managing and improving the career path selection process for students. It interacts with users through a user-friendly web interface. It may also integrate with external databases and services for enhanced functionality.

2.2 Product Features

The software shall provide the following key features:

2.2.1 User Registration and Login

- Users can register for an account.
- Users can log in using their credentials.

2.2.2 Career Path Analysis

- The system shall analyze user choices and interests.
- It shall provide advice on the best career options for the user.
- It shall determine suitability for a specific career or role type.

2.2.3 Form Submission

- Users can fill out a form providing their information, including their full name, class, subject, graduation, and interests.

- The system shall process this data for career path analysis.

2.2.4 University Information

- Display information about universities, including logos and details.

2.3 User Classes and Characteristics

The software is primarily designed for two user classes:

- Students: Users who are looking for career guidance.
- Administrators: Users who manage the system's content and user data.

3. Specific Requirements

3.1 User Interface Requirements

3.1.1 Registration and Login

- Users shall be able to register with a valid email address and password.
- Users shall be able to log in with their registered credentials.

3.1.2 Form Page

- Provide a user-friendly form page for inputting personal information.
- Use JavaScript to show the user's name after form submission.

3.1.3 Results Page

- Display analysis results based on user inputs.

- Show recommended courses, entrance exams, and top colleges.

3.3 Non-Functional Requirements

3.2.1 Performance

- The system shall respond to user interactions within 2 seconds.
- It shall handle up to 1000 concurrent users without performance degradation.

3.2.2 Security

- User passwords shall be securely stored and hashed in the database.
- Sensitive user data shall be protected from unauthorized access.

3.2.3 Compatibility

- The system shall be compatible with the latest versions of major web browsers (Chrome, Firefox, Safari, Edge).

3.2.4 Usability

- The user interface shall be intuitive and easy to use, ensuring a positive user experience.

4. External Interfaces

4.1 Database

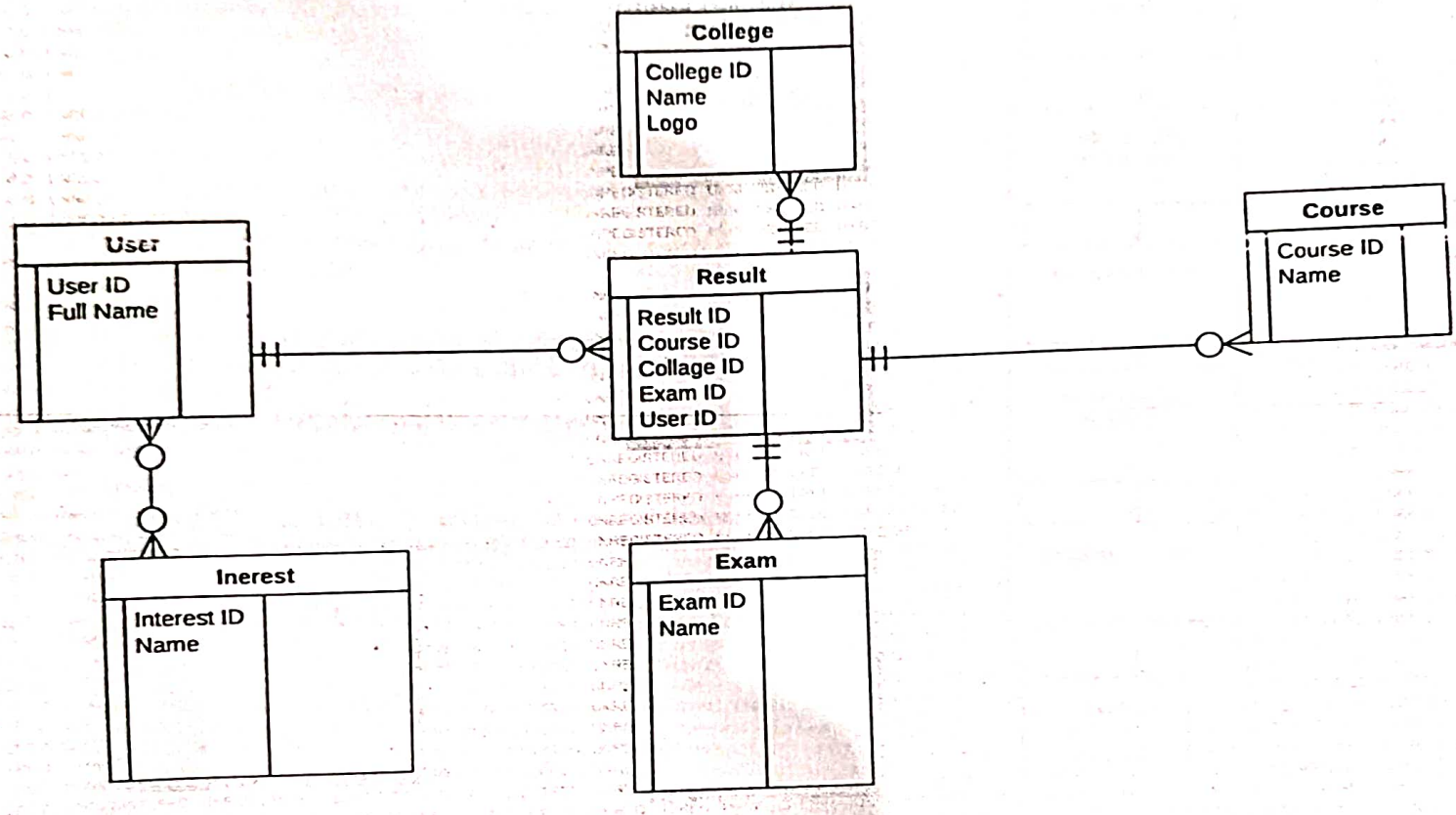
- The software shall interact with a database to store user information and analysis results.

5. Constraints

- The system should be developed using HTML, CSS, JavaScript for the front-end and a backend technology of choice (e.g., Node.js, Python, Ruby) for server-side processing.

- The system should comply with data protection and privacy regulations.

Data Model1::ERDDiagram1



Flowchart1::FlowchartDiagram1

1-User Input Form:

You need to create an HTML form to collect user data, such as their name, class (education level), subject (if applicable), graduation, and interests.

When the user submits the form, you need to store this information, possibly in local storage.

2-Analysis of User Data:

You need to create JavaScript logic that analyzes the user's input data to determine suitable career options.

This analysis may involve conditional statements to match user data with predefined career paths.

3-Display Results:

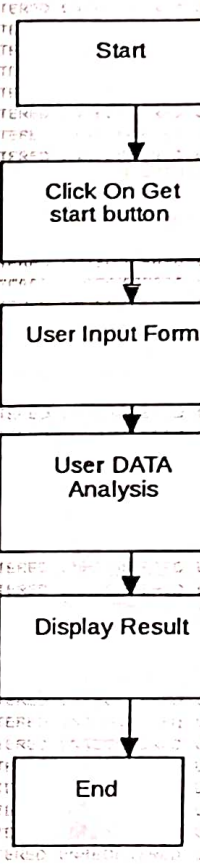
Once the analysis is complete, you should display the recommended career paths, entrance exams, and colleges that match the user's profile.

You can update the HTML document with this information using JavaScript.

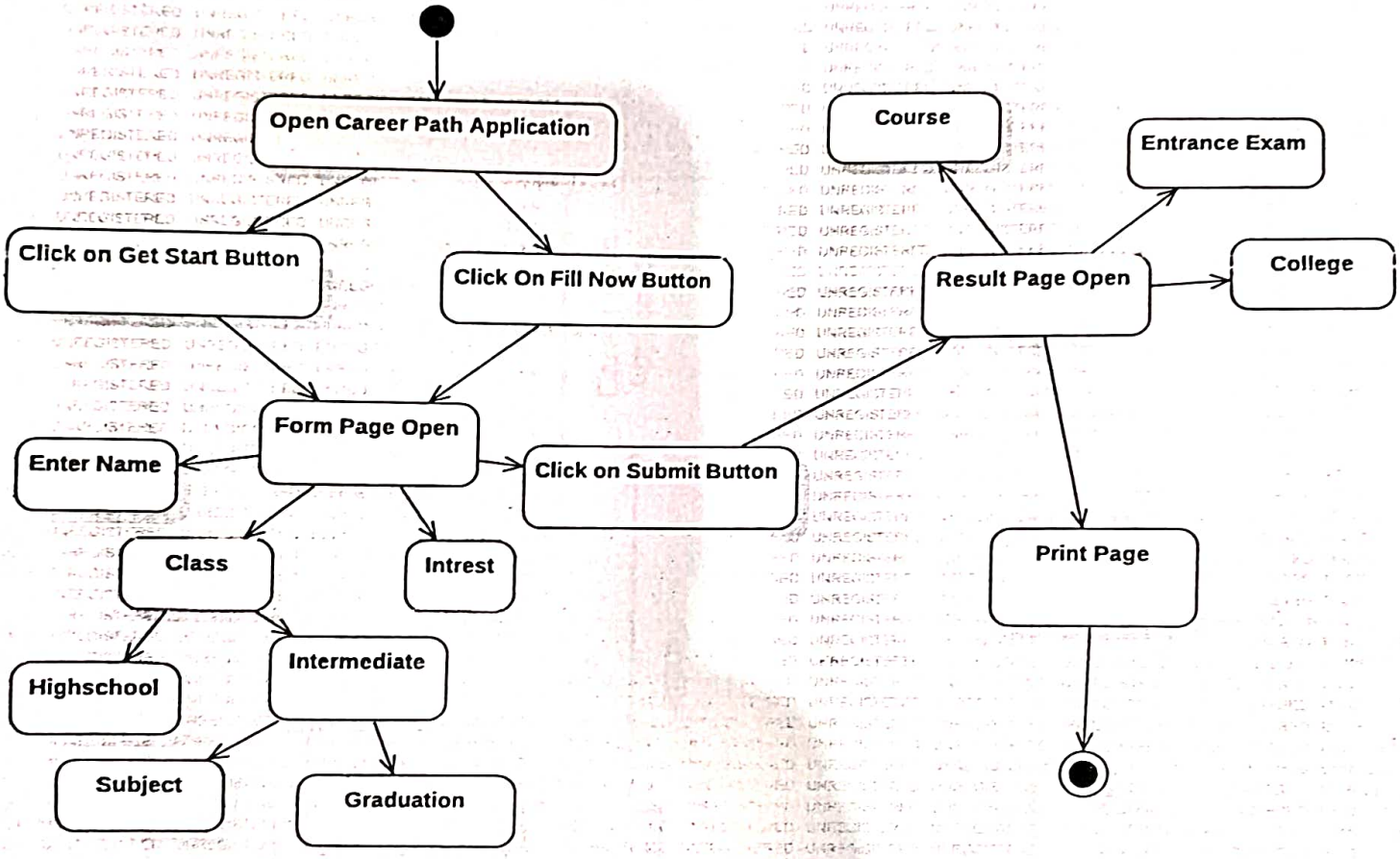
4-Print Results:

You can provide a "Print" button to allow users to print their results.

This can be done using the window.print() function in JavaScript.



Activity1::ActivityDiagram1



DataFlowModel1::DFDDiagram1