System and Software Architecture Description (SSAD)

Swim Meet Sign-Up

Team 03

Member	Role
Archan Dutta	Project Manager, Life Cycle Planner, Tester
Swasti Sharma	Operational Concept Engineer
Rasleen Sahni	Feasibility Analysist, Website Maintainer
Deepanshu Suneja	Software Architect
Vibhanshu Sharma	Software Prototyper
Jenny Greer	IIV&V, Quality Management Focal Point,

December 2, 2017

Version History

Date	Author	Version	Changes made	Rationale
11/12/17	DS	1.1	• Original template for use with Instructional ICM-Sw v1.0	• Initial draft for use with Instructional ICM-Sw v1.0
12/02/17	JG	2.0	• Updated format to meet current template, fixed multiple diagrams	• To be complete as possible as part of TRR

Table of Contents

Table of Tables

Table of Figures

1. Introduction

1.1 Purpose of the SSAD

The purpose of the SSAD is to analyze and design the details about the system software, architecture and hardware parts that will be used in the project. This report presents the logical architecture of the system and the key properties of the system by analyzing the system context diagram and showing the use cases.

1.2 Status of the SSAD

This is the second version of the SSAD. This document was updated to match the Architected Agile Template. Also corrected feedback we received from the graders during from the DCP Package. This is being updated and reviewed in preparation for TRR.

2. System Analysis

2.1 System Analysis Overview

The primary purpose of our system is to provide a web platform for parents to sign up swimming events for their kids. The system is a technological advancement of the current manual system.

The current system has been identified as a time consuming and unnecessary effort utilizing process. The new system will allow the head coach of Arcadia Reptiles to upload and extract events from PDF document provide by the USA Swimming Association. The parents will have the ease of Signing up events with their kid's information whereas the head coach will have the ease of generating a comprehensive report consisting of all the signed-up events.

2.1.1 System Context

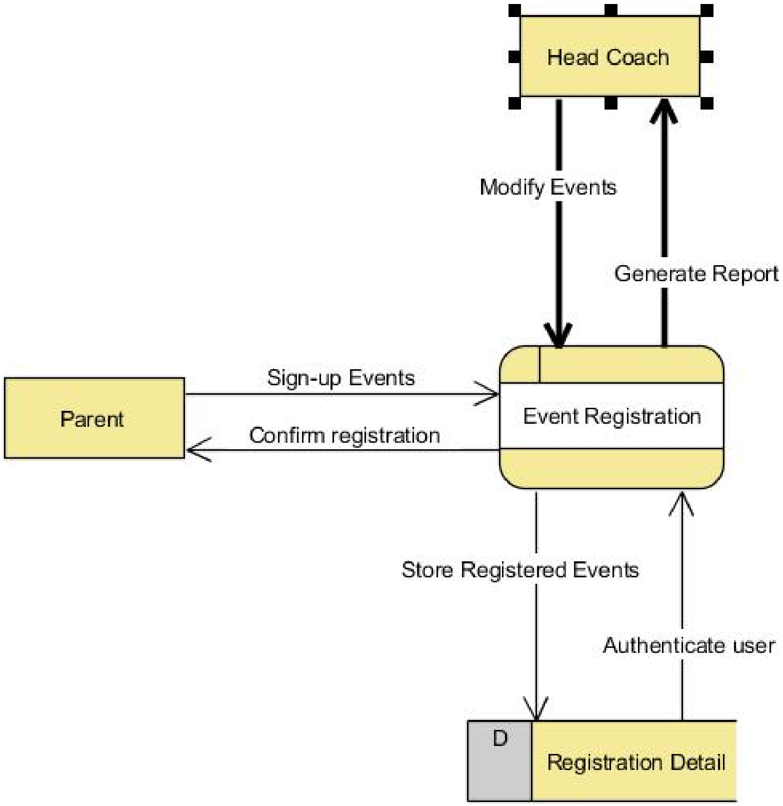

Figure 1: System Context Diagram

Table 1: Actors Summary

Actor	Description	Responsibilities
Parent	User who will sign up events for their kids.	Login with their kid's registration ids and Sign-up events before the deadline provided.
Head Coach	User who will be the providing the event info	Upload and extract PDF to add event information for every month

2.1.2 Artifacts & Information

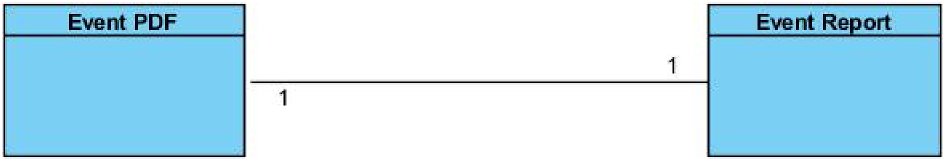

Figure 2: Artifacts and Information Diagram

Table 2: Artifacts and Information Summary

Artifact	Purpose
Swim Meet PDF	PDF document provided by the USA Swimming Association to the Head Coach every month.
Swim Event Participation Report	Report consisting of all the signed-up events and corresponding kids information, generated by the Head Coach.

2.1.3 Behavior

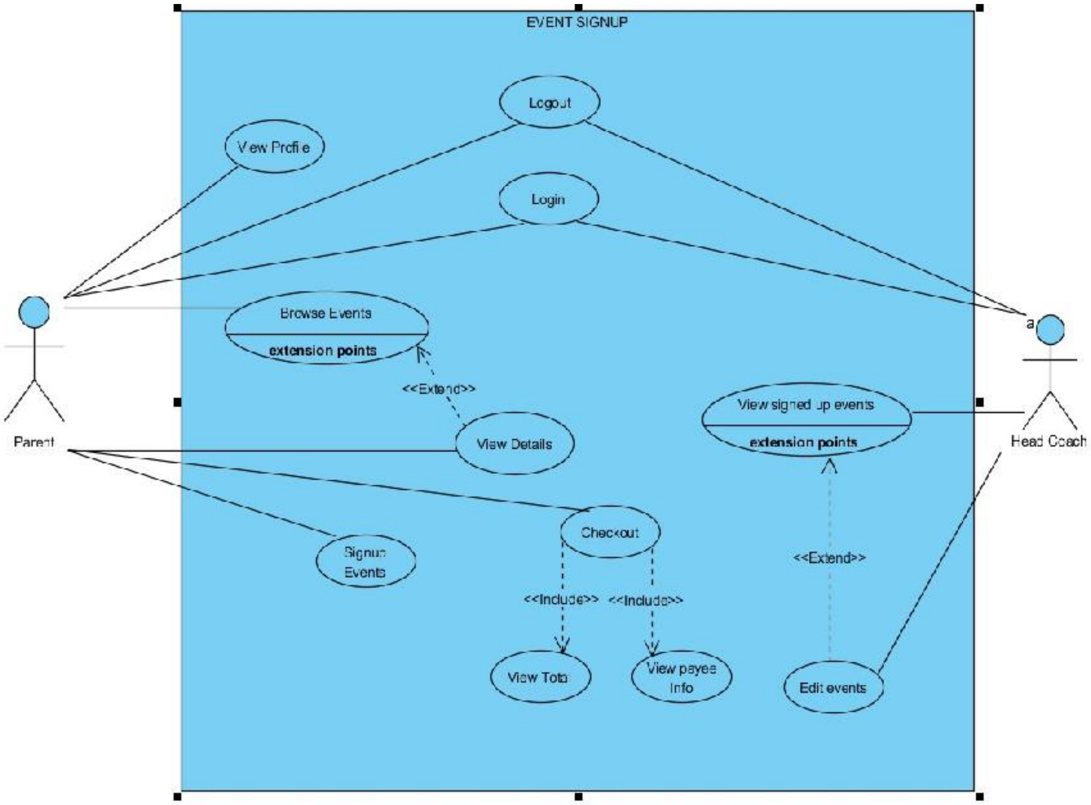

Figure 3: Process Diagram for Event Signup

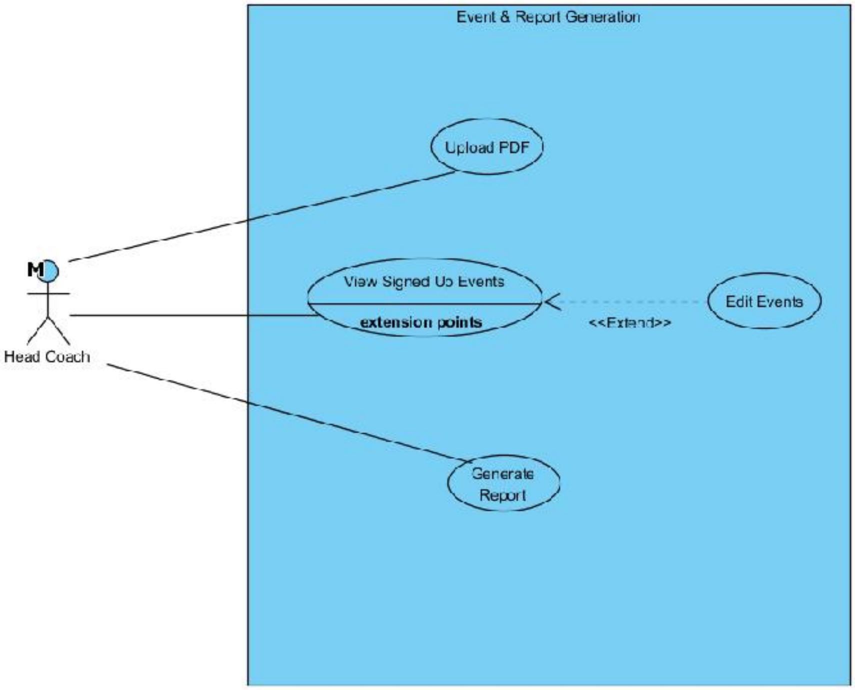

Figure 4: Process Diagram for Event and Report Generation

2.1.3.1 Capability Event Signup

2.1.3.1.1 Process Login

Table 3: Login Process Description

Identifier	UC-1: Login
Purpose	Authorizing a user to log into the system as either a Head Coach or Parent
Requirements	User (Head Coach/Parent) specific Login
Development Risks	Secure connection protocol may be required.
Pre-conditions	The users are on login page
Post-conditions	The user is logged in as either a Head Coach or a Parent. Unauthorized users are denied.

Table 4: Login Typical Course of Action

Seq#	Actor’s Action	System’s Response
1	Enter username and password	
2	Click Login Button	

3		Verify username and password is in the database
4		Authorize the user
5		Redirect the user to Head Coach/Parent page

Table 5: Login Exceptional Course of Action

Seq#	Actor's Action	System's Response
1	Enter username and password	
2	Click Login button	
3		Verify username and password
4		The username/password pair is not in the database; show error message.

2.1.3.1.2 Process Logout

Table 6: Logout Process Description

Identifier	UC-2 Logout
Purpose	User to Logout of Head Coach/Parent website so that others can't access their information
Requirements	Logout of website
Development Risks	None
Pre-conditions	The user has logged into the system as either a Head Coach or Parent.
Post-conditions	The Head Coach/Parent logged out, and the session is terminated.

Table 7: Logout Typical Course of Action

Seq#	Actor's Action	System's Response
1	Click the Logout button	
2		Log the user out and deactivate the user's session
3		Redirect the user to the log in page

2.1.3.1.3 Process View Profile

Table 8: View Profile Process Description

Identifier	UC-3: View Profile
Purpose	Provide registered information about the user

Requirements	View users information about Parent and Child
Development Risks	None
Pre-conditions	User clicks in profile button
Post-conditions	User views their information

Table 9: View Profile Typical Course of Action

Seq#	Actor's Action	System's Response
1	Enter Profile Page	
2		Retrieve a list of the Parents User Information
3		Display the Parents User Information which should include child's name, child's age, past swim event

2.1.3.1.4 Process Signup Events

Table 10: Signup Events Process Description

Identifier	UC-4 Signup Events
Purpose	For the parent to select swim events for their child to sign-up for that is at least one but less than the Swim Meet maximum
Requirements	Parent should be able to sign up their child for swim events
Development Risks	None
Pre-conditions	Parent is logged into website. Swim Event data exists
Post-conditions	Parent's Child is signed up for various swim events

Table 11: Signup Events Typical Course of Action

Seq#	Actor's Action	System's Response
1	Parent select swim meet page	
2		Retrieve available swim meets that have not expired
3	Parent selects desired swim meet by date	
4		Compare Childs age to Swim events the child can participate it
5		Display available Swim events
6	Parent selects desired swim event	
7		Checks that the max number of event allowable to participate in has not been reached
8		Swim Event is added to cart

Table 12: Signup Events Exceptional Course of Action

Seq#	Actor's Action	System's Response
1	Parent select swim meet page	
2		Retrieve available swim meets that have not expired
3	Parent selects desired swim meet by date	
4		Compare Childs age to Swim events the child can participate it
5		Display available Swim events
6	Parent selects desired swim event	
7		Checks that the max number of event allowable to participate in has not been reached
8		Identify that max number of swim events has been reached and report error message to Parent
9		Swim Event is not added to cart

2.1.3.1.5 Process Browse Events

Table 13: Browse Events Process Description

Identifier	UC-5: Browse Events
Purpose	Parents and head coach can view all applicable swim event for all available swim meets
Requirements	Able to view available swim events
Development Risks	None
Pre-conditions	User (Head Coach/Parent) is logged into website
Post-conditions	Able to see swim events

Table 14: Browse Events Typical Course of Action

Seq#	Actor's Action	System's Response
1	User select swim meet page	
2		Retrieve available swim meets that have not expired
3	User selects desired swim meet by date	

4		Compare user credentials to available swim events and determine which ones can be displayed
5		Display available Swim events

2.1.3.1.6 Process View Details

Table 15: View Details Process Description

Identifier	UC-6: View Details
Purpose	Users are able to view the details of a specific swim event
Requirements	Users shall be able to see information about a swim event including age requirement, distance, and date
Development Risks	None
Pre-conditions	Users are logged into website
Post-conditions	Users are to view details about a specific swim event

Table 16: View Details Typical Course of Action

Seq#	Actor's Action	System's Response
1	User select swim meet page	
2		Retrieve available swim meets that have not expired
3	User selects desired swim meet by date	
4		Compare user credentials to available swim events and determine which ones can be displayed
5		Display available Swim events
6	User selects desired swim event	
7		Retrieve details about the swim event

2.1.3.1.7 Process Checkout

Table 17: Checkout Process Description

Identifier	UC-7: Checkout
Purpose	Parent is able to confirm desired selection of swim events they want their child to participate in
Requirements	Parent signs-up their child for specific swim events
Development Risks	None
Pre-conditions	Parent has selected swim events in their cart
Post-conditions	Parents can see their child is confirmed to participate in specific events

Table 18: Checkout Typical Course of Action

Seq#	Actor's Action	System's Response
1	Parent selects checkout button	
2		Saves child's information as one who will be participating in swim event
3		Shows Confirmation of events child is signed up for

2.1.3.1.8 Process View Total

Table 19: View Total Process Description

Identifier	UC-8: View Total
Purpose	Parent able to see total cost of all swim events that they have signed their child up for
Requirements	Parents shall know how much to pay to have their child participate in a specific swim meet
Development Risks	None
Pre-conditions	Parent has selected checkout button
Post-conditions	Cost is provided to Parent

Table 20: View Total Typical Course of Action

Seq#	Actor's Action	System's Response
1	Parent selects checkout button	
2		Sums the cost of all events
3		Displays total cost of all events

2.1.3.1.9 Process View Payee Information

Table 21: View Payee Information Process Description

Identifier	UC-9: View Payee Information
Purpose	The parent knows who to make check out to so they can pay for their child to participate in swim event
Requirements	Parent shall know who to pay
Development Risks	None
Pre-conditions	Parent has checkout with at least 1 swim event
Post-conditions	Payment information is viewed

Table 22: View Payee Information Typical Course of Action

Seq#	Actor's Action	System's Response

1	Parent selects checkout button	
2		Retrieves specific payment information for Swim Meet
3		Displays payment information

2.1.3.1.10 Process View Signed up Events

Table 23: View Signed up Events Process Description

Identifier	UC-10 View Signed up Events
Purpose	Provide user the functionality to look up all eligible events
Requirements	Extracted events from the PDF
Development Risks	Mismatch in the eligibility criteria
Pre-conditions	User logs in with their registration id
Post-conditions	Events are displayed with their details

Table 24: Typical Course of Action

Seq#	Actor's Action	System's Response
1	User selects view signed up events	
2		Retrieve events signed up for
3		Display information to the user

2.1.3.1.11 Process Edit Event Signups

Table 25: Edit Event Signups Process Description

Identifier	UC 11: Edit Event Signups
Purpose	Head Coach can change or update information about specific swim events
Requirements	Head Coach shall be able to edit event information
Development Risks	None
Pre-conditions	Swim events exist and children are signed up for a specific swim event
Post-conditions	Swim event information is changed

Table 26: Edit Event Signups Typical Course of Action

Seq#	Actor's Action	System's Response
1	Head Coach selects a specific swim event	
2	Head Coach selects the view details button	

3		Retrieve swim event details
4	Head Coach selects a parameter to change	
5		Opens a text box for editing
6	Head Coach inputs new parameter	
7	Head Coach Selects save	
8		Event details are udated
9		Updated details are displayed

2.1.3.2 Capability Event and Report Generation

2.1.3.2.1 Process PDF Uploading and Parsing PDF

Table 27: PDF Upload Process Description

Identifier	UC-12: PDF Upload
Purpose	Providing the source of event information so that Parents Can sign their children up for activities
Requirements	Parse a PDF file provided by the USA Swimming Association.
Development Risks	1. Change in PDF format 2. Extract incorrect event information
Pre-conditions	Have a pdf file in an acceptable format and head coach is logged in
Post-conditions	Head coach can see the extracted events.

Table 28: PDF Upload Typical Course of Action

Seq#	Actor's Action	System's Response
1	Head Coach selects the Upload PDF page	
2		Retrieve the upload page
3	Head Coach selects the upload PDF button	
4		Opens a dialog box to allow coach to select file
5	Head Coach selects file for uploading	
6		Retrieves file
7		Parses file to extract data
8		Puts data into database
9		Displays data extracted

2.1.3.2.2 Process Generate Report

Table 29: Generate Report Process Description

Identifier	UC-13: Generate Report
Purpose	Generating file containing aggregated data of signed up events
Requirements	Head Coach shall be able to generate a report of all children signed up for a swim event
Development Risks	None
Pre-conditions	Children are signed up for swim events in a given swim meet.
Post-conditions	Head coach can download the pdf and csv file.

Table 30: Generate Report Typical Course of Action

Seq#	Actor's Action	System's Response
1	Head coach Selects Generate report button	
2		Display dialog box to ask which file type to generate report in
3	Head Coach selects desired file type	
4		Retrieves information needed for report
5		Formats information for specific file type
6		Creates the file in the specified file type
7		Displays file to user in new window

2.1.4 Modes of Operation

Not Applicable. There is only one mode of operation.

2.2 System Analysis Rationale

The Website being developed for Arcadia Riptides Swim Club has 2 levels of access control. This control is based on external users using the system.

1. **Parent**: Parents will sign-up their kids for the Swimming Events. Parents can see the events that their kid is participating in. They cannot alter information about the Swimming Events.

2. **Head-Coach:** The Head-Coach will upload PDF containing information about the Swim Meet and its Events. He/She can set Deadline for the events and alter the Signups.

3. Technology-Independent Model

3.1 Design Overview

3.1.1 System Structure

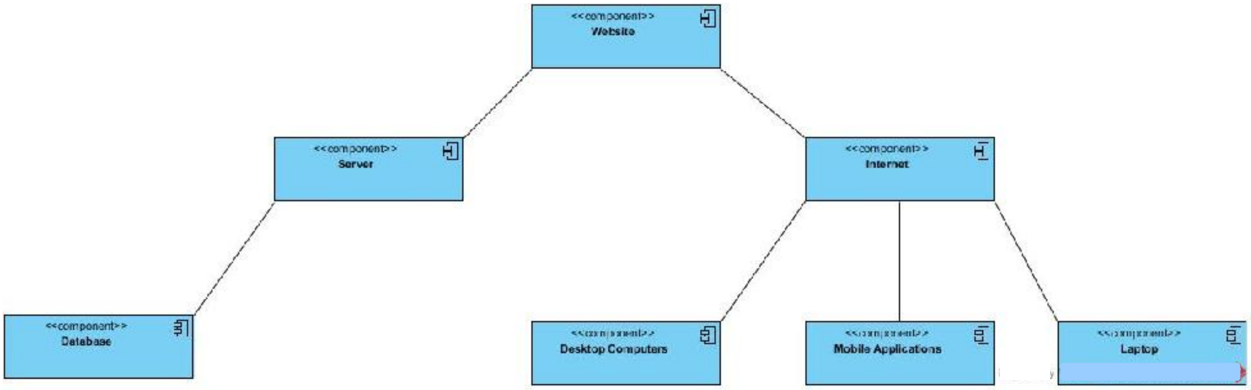

Figure 5: Conceptual Domain Model

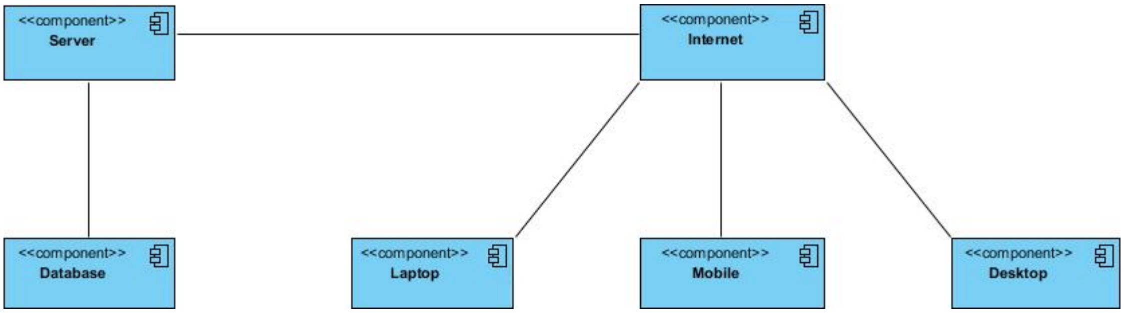

Figure 6: Hardware Component Class Diagram

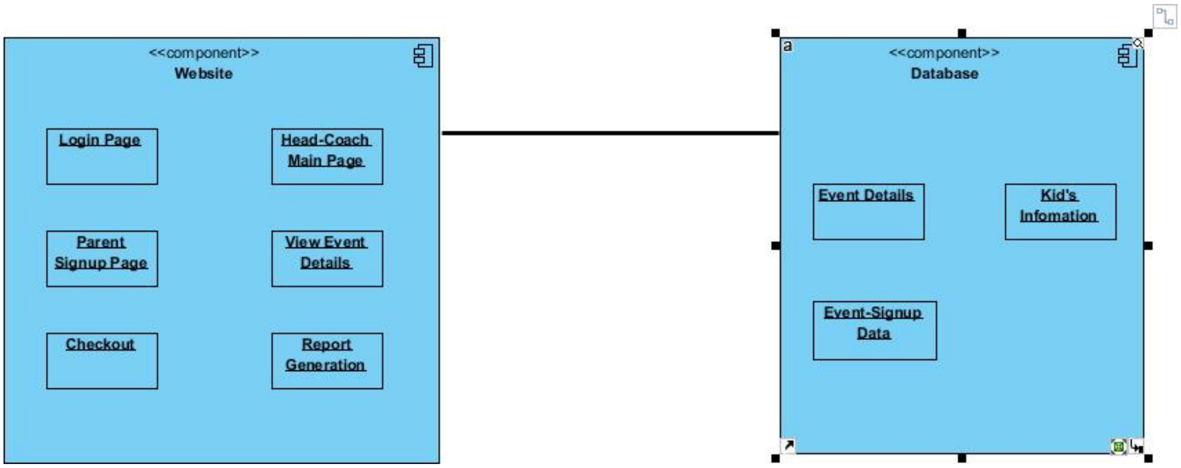

Figure 7: Software Component Class Diagram

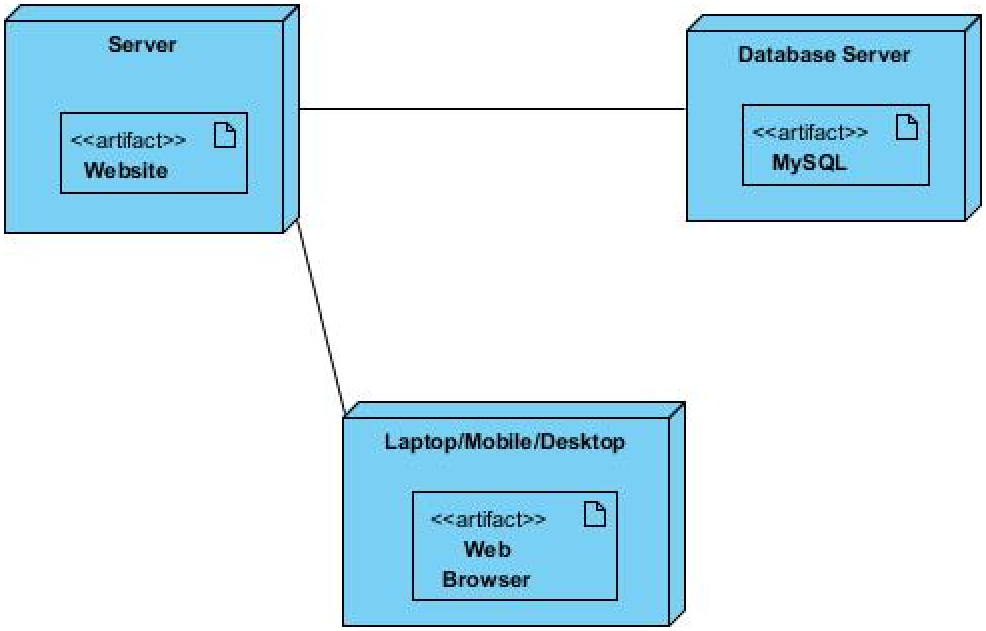

Figure 8: Deployment Diagram

Table 31: Hardware Component Description

Hardware Component	Description
Server	The website is deployed on the server
Internet	A medium of connection between website and user
Database	A database storing information about events, kids and signups
Mobile	Device to access the website
Desktop	Device to access the website
Laptop	Device to access the website

Table 32: Software Component Description

Software Component	Description
Login Page	Login page for Head-Coach and Parent
Parent Signup Page	Parent can sign-up/register their kid for the event
Checkout	See the total amount to be paid based on sign-ups
Head-Coach Main Page	Upload PDF for a swim meet
View Event Details	Display Information about the events
Report Generation	

3.1.2 Design Classes

3.1.2.1 Swim Meet Website Design

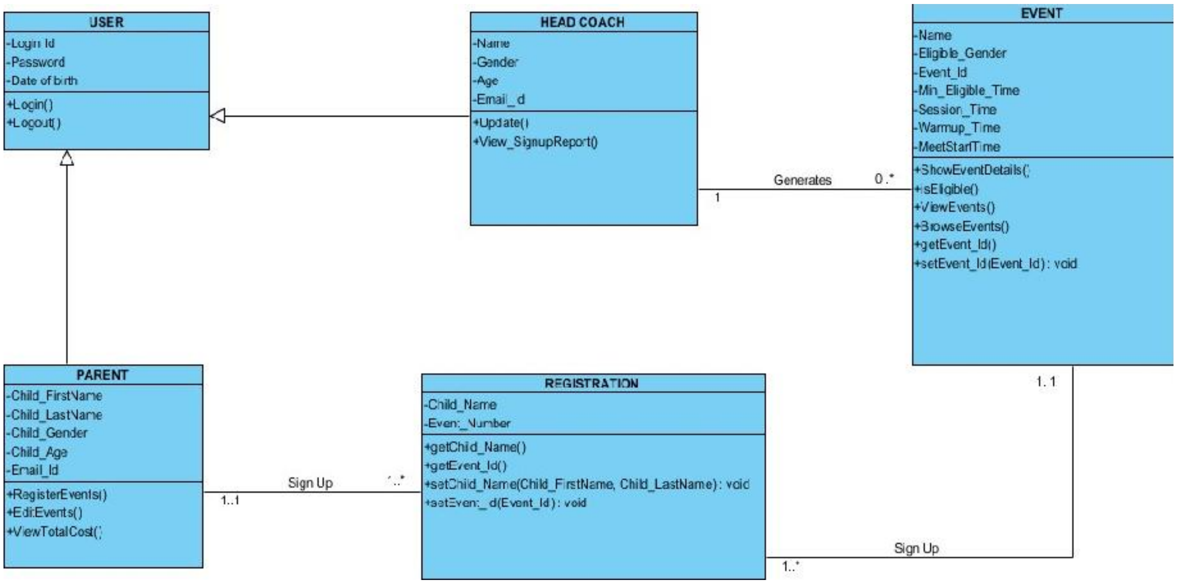

Figure 9: Design Class Diagram

Table 33: Design Class Description

Class	Type	Description
User	Entity	Have access to the system
Head Coach	Entity	Can Upload PDF, edit events, signups/registrations
Parent	Entity	Signup their kid for their swimming events
Event	Entity	The Swimming Events in a meet
Registration	Component	Signups made by the Parent

3.1.3 Process Realization

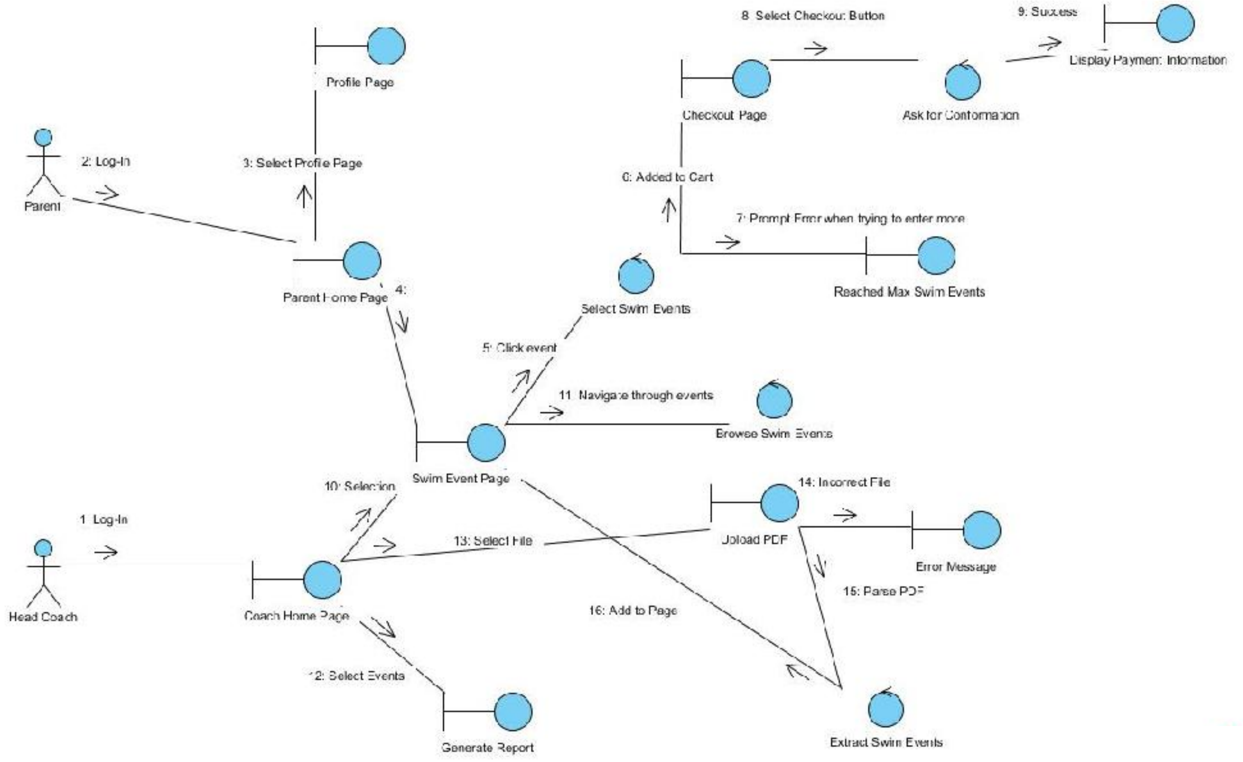

Figure 10: Robustness Diagram

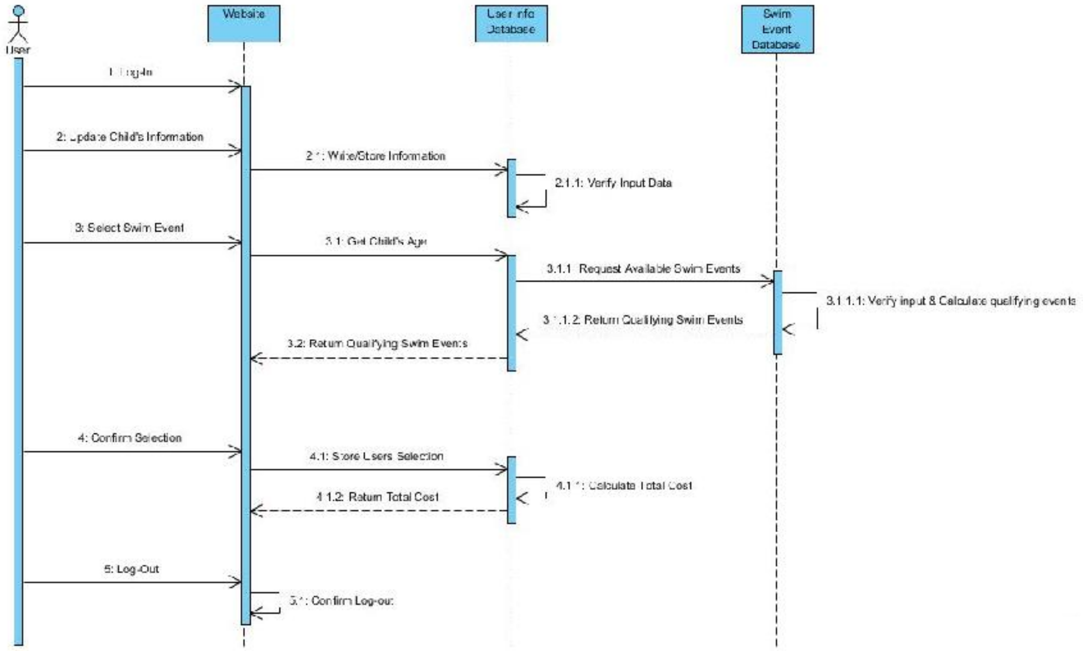
User
Website
User Info Database
Swim Event Database
1: Log-In
2: Update Child's Information
2.1: Write/Store Information
2.1.1: Verify Input Data
3: Select Swim Event
3.1: Get Child's Age
3.1.1: Request Available Swim Events
3.1.1.1: Verify input & Calculate qualifying events
3.1.1.2: Return Qualifying Swim Events
3.2: Return Qualifying Swim Events
4: Confirm Selection
4.1: Store Users Selection
4.1.1: Calculate Total Cost
4.1.2: Return Total Cost
5: Log-Out
5.1: Confirm Log-out

Figure 11: Sequence Diagram Parent Event Signup

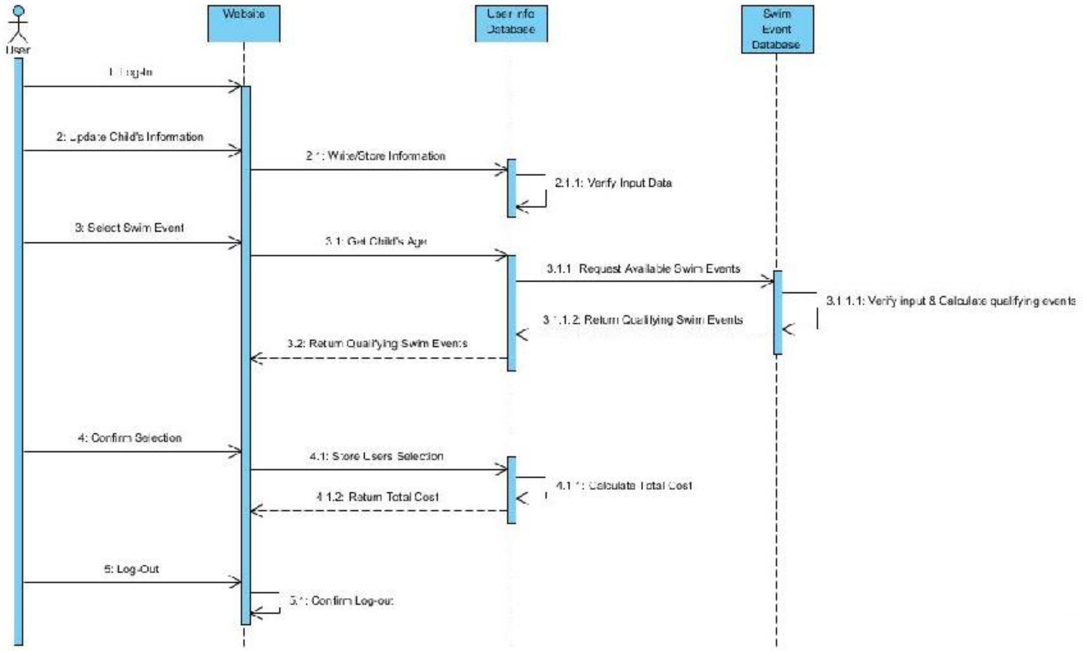
User
Website
User Info Database
Swim Event Database
1: Log-In
2: Update Child's Information
2.1: Write/Store Information
2.1.1: Verify Input Data
3: Select Swim Event
3.1: Get Child's Age
3.1.1: Request Available Swim Events
3.1.1.1: Verify input & Calculate qualifying events
3.1.1.2: Return Qualifying Swim Events
3.2: Return Qualifying Swim Events
4: Confirm Selection
4.1: Store Users Selection
4.1.1: Calculate Total Cost
4.1.2: Return Total Cost
5: Log-Out
5.1: Confirm Log-out

Figure 12: Sequence Diagram Head Coach Report Generation

3.2 Design Rationale

The Architecture of the website is based on the Win-Win conditions from the negotiations with the client so that the website performs its functions efficiently. Several Open Source Softwares have been used to reduce cost of development. The two users (Parent and Head-Coach) will follow the above mentioned sequences to achieve their desired output from the website. The database will adhere strongly to the class diagram.

4. Technology-Specific System Design

4.1 Design Overview

4.1.1 System Structure

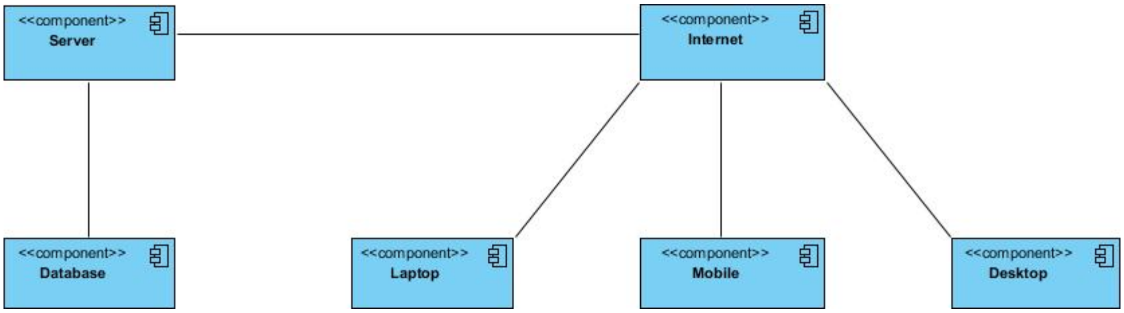

Figure 13: Hardware Component Class Diagram

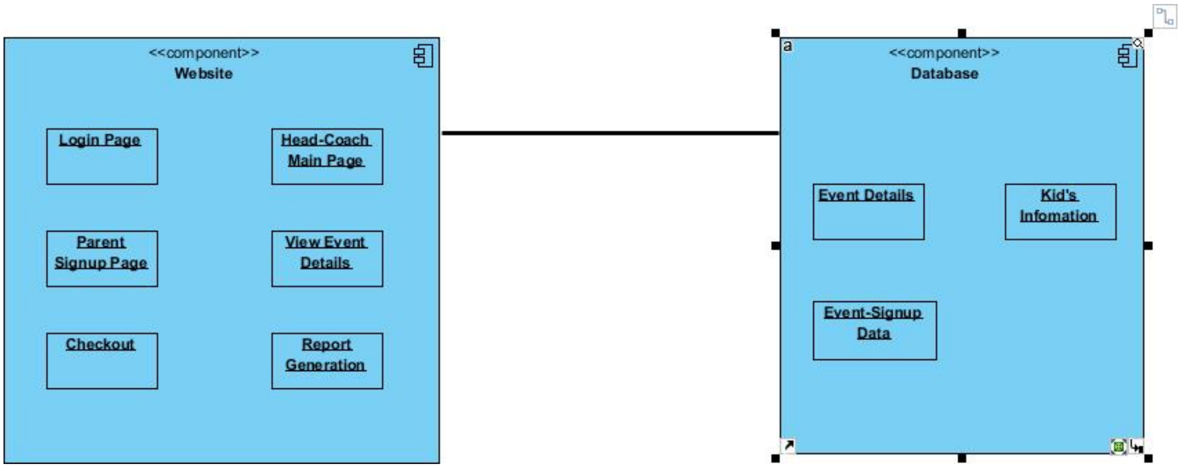

Figure 14: Software Component Class Diagram

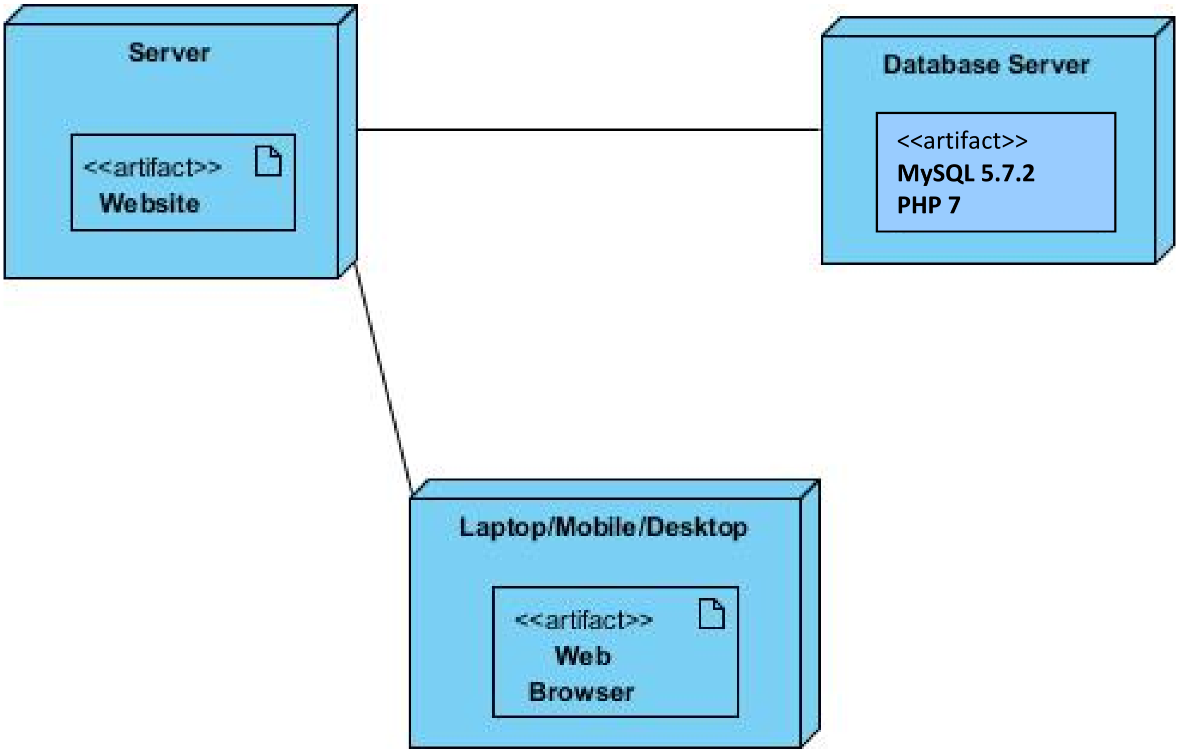

Figure 15: Deployment Diagram

Table 34: Hardware Component Description

Hardware Component	Description
Server	The website is deployed on the server
Internet	A medium of connection between website and user
Database	A database storing information about events, kids and signups
Mobile	Device to access the website
Desktop	Device to access the website
Laptop	Device to access the website

Table 35: Software Component Description

Software Component	Description
Login Page	Login page for Head-Coach and Parent
Parent Signup Page	Parent can sign-up/register their kid for the event
Checkout	See the total amount to be paid based on sign-ups
Head-Coach Main Page	Upload PDF for a swim meet
View Event Details	Display Information about the events
Report Generation	Create Report of Kids registered for an event

4.1.2 Design Classes

4.1.2.1 Swim Meet Website Design Classes

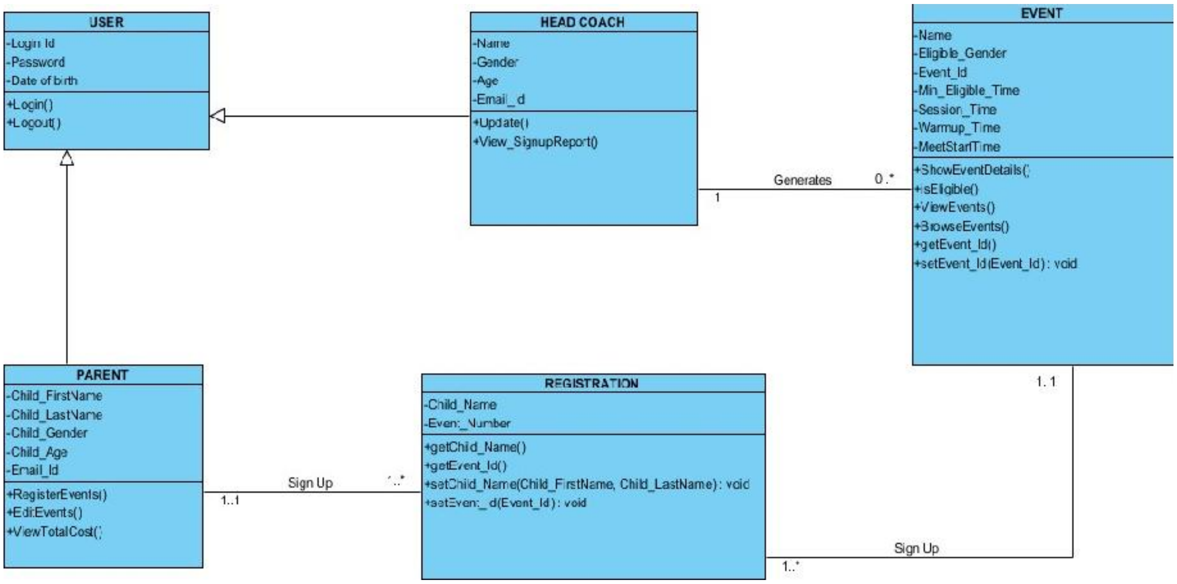

Figure 16: Swim Meet Design Class Diagram

Table 36: Swim Meet Design Class Description

Class	Type	Description
User	Entity	Have access to the system
Head Coach	Entity	Can Upload PDF, edit events, signups/registrations
Parent	Entity	Signup their kid for their swimming events
Event	Entity	The Swimming Events in a meet
Registration	Component	Signups made by the Parent

4.1.3 Process Realization

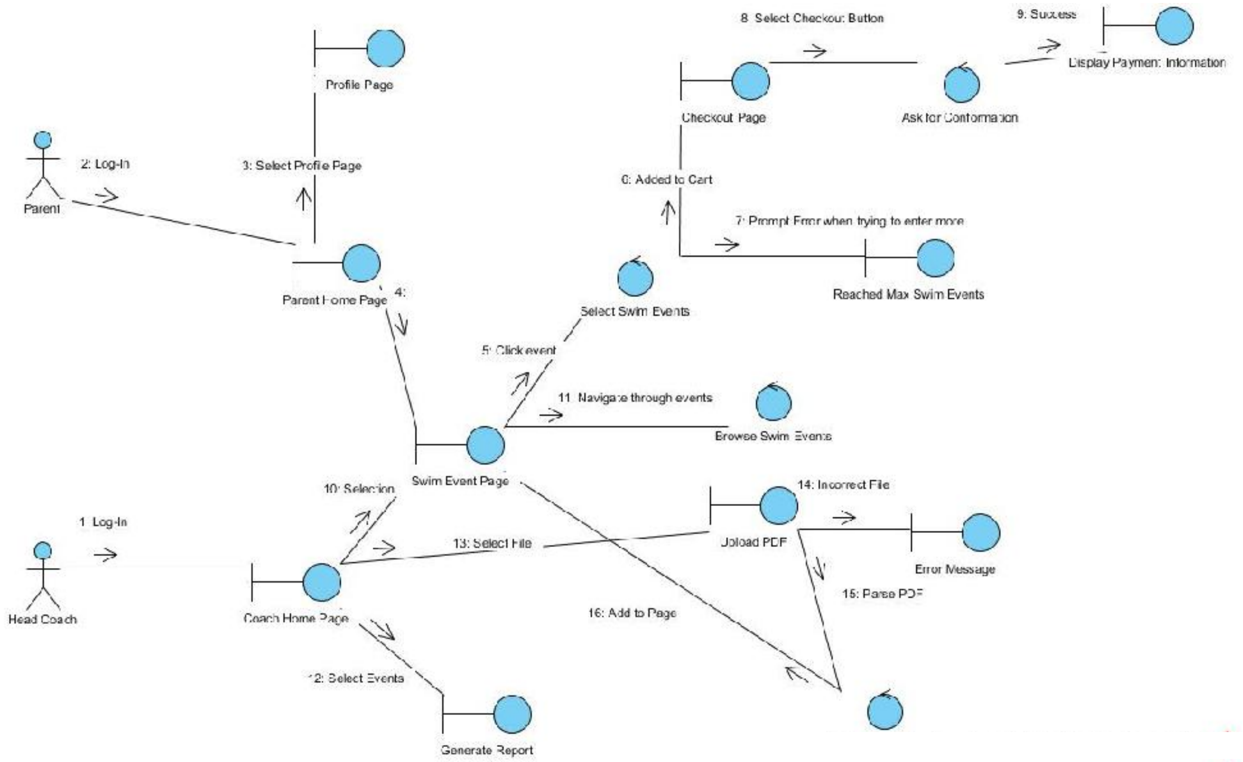

Figure 17: Robustness Diagram

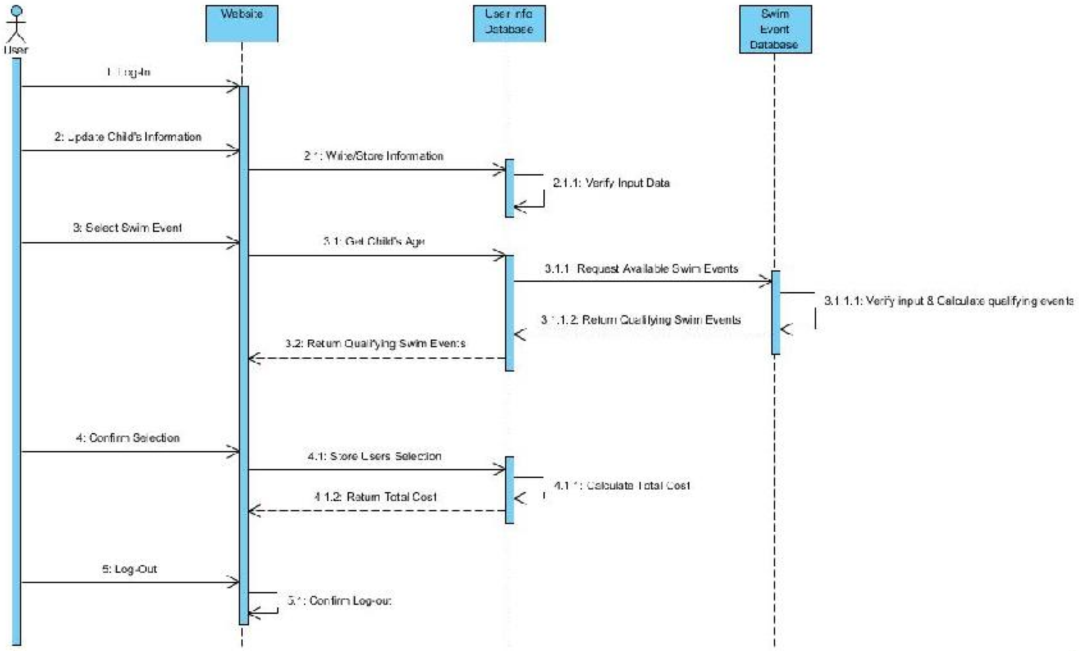
User
Website
User Info Database
Swim Event Database
1: Log-In
2: Update Child's Information
2.1: Write/Store Information
2.1.1: Verify Input Data
3: Select Swim Event
3.1: Get Child's Age
3.1.1: Request Available Swim Events
3.1.1.1: Verify input & Calculate qualifying events
3.1.1.2: Return Qualifying Swim Events
3.2: Return Qualifying Swim Events
4: Confirm Selection
4.1: Store Users Selection
4.1.1: Calculate Total Cost
4.1.2: Return Total Cost
5: Log-Out
5.1: Confirm Log-out

Figure 17: Sequence Diagram Parent Event Signup

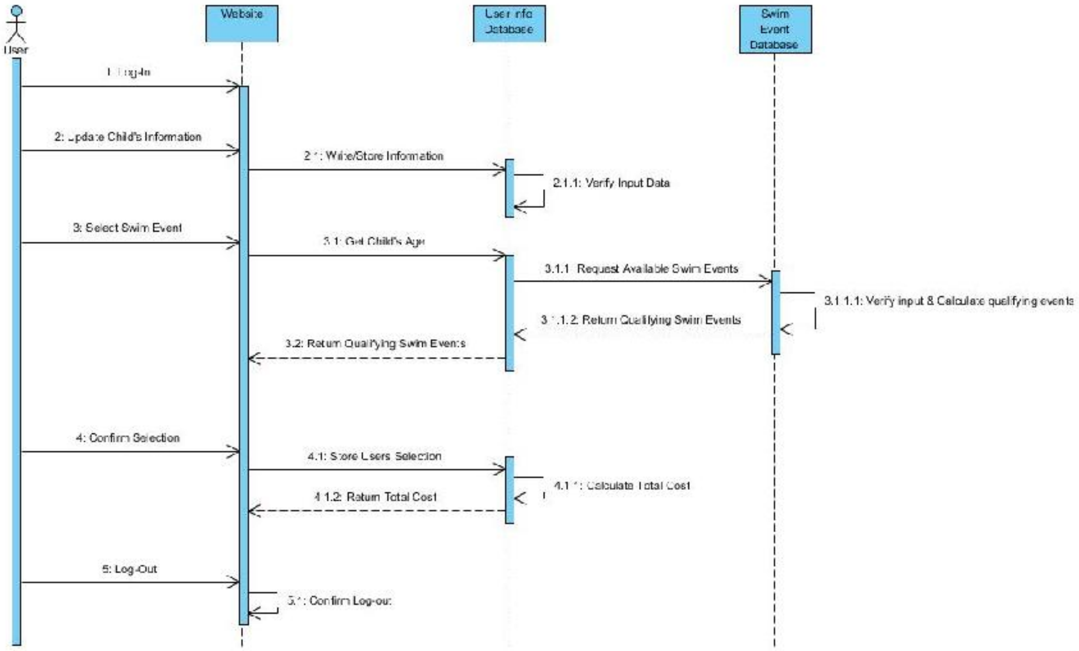
User
Website
User Info Database
Swim Event Database
1: Log-In
2: Update Child's Information
2.1: Write/Store Information
2.1.1: Verify Input Data
3: Select Swim Event
3.1: Get Child's Age
3.1.1: Request Available Swim Events
3.1.1.1: Verify input & Calculate qualifying events
3.1.1.2: Return Qualifying Swim Events
3.2: Return Qualifying Swim Events
4: Confirm Selection
4.1: Store Users Selection
4.1.1: Calculate Total Cost
4.1.2: Return Total Cost
5: Log-Out
5.1: Confirm Log-out

Figure 18: Sequence Diagram Head Coach Report Generation

4.2 Design Rationale

The Architecture of the website is based on the Win-Win conditions from the negotiations with the client so that the website performs its functions efficiently. Several Open Source Softwares have been used to reduce cost of development. The two users (Parent and Head-Coach) will follow the above mentioned sequences to achieve their desired output from the website. The database will adhere strongly to the class diagram.

5. Architectural Styles, Patterns and Frameworks

Table 37: Architectural Styles, Patterns, and Frameworks

Name	Description	Benefits, Costs, and Limitations
PHP	Framework designed for Server Side Scripting	• Widely used and therefore great documentation • Open Source and Free • Stable and Easy to Use • Not suitable for large applications
AngularJS	Front end framework used in Conjunction with HTML and CSS	• Open Source and Free • Responsive • Fast Development • Not Easy to Learn
Client Server Architecture Style	The client–server model is a distributed application structure that partitions tasks or workloads between the providers of a resource or service, called servers, and service requesters, called clients.	• Easy Maintenance • Security • Improved Data Sharing • Suitable for Web applications such as this project • Prone to service disruption because it is centralized
Pattern : 3-Tier	System divided into Front-tier, Middle-tier, Backend-tier	• Highly Modular • Easy to maintain and understand code • Increased complexity
Observer Design Pattern	The observer pattern is a software design pattern in which an object, called the subject, maintains a list of its dependents, called observers, and notifies them automatically of any state changes, usually by calling one of their methods.	• Supports the principle of loose coupling between objects that interact with each other. • Allows sending data to other objects effectively without any change in the Subject or Observer classes. • Observers can be added/removed at any point in time.