Project Report : Live Scores

Course Code : CSE 2112

Supervisor

Monisha Dey

Lecturer,Department of Computer Science & Engineering

Mymensingh Engineering College

Submitted By

Md. Shakil Anower Samrat (Roll: 307, Reg:2850)

Md. Jannatul Naeem(Roll:338, Reg:2880)

Tajnim Jaman Mim(Roll:356, Reg:2898)

Md. Shahab Uddin (Roll: 318, Reg:2861)

Md. Sujon Ahamed(Roll:328 ,Reg:2870)

Session:2019-20

Department of Computer Science and Engineering

Mymensingh Engineering College

Submission Date : 25 July,2022

Statement

We now state that the project "LiveScores" presented in full compliance of the criteria for "Java Project" is wholly our own work and includes no content borrowed from others, save where proper reference is included in the report.

Sincere Regards,

— — — — — — — — — — — — — — –

Md. Shakil Anower Samrat

Roll: 309

Reg: 2852

Md. Jannatul Naeem

Roll: 338

Reg: 2880

Tajnim Jaman Mim

Roll: 356

Reg: 2898

Md. Shahab Uddin

Roll: 318

Reg: 2861

Md. Sujon Ahamed

Roll: 328

Reg: 2870

Session : 2019-2020

Department of Computer Science & Engineering

Mymensingh Engineering College

Approval

The project "Livescores" presented by Md. Shakil Anower Samrat, Tajnim Jaman Mim, Md. Jannatul Naeem, Md. Shahab Uddin, and Md. Sujon Ahamed under the supervision of Ms. Monisha Dey in the Department of Computer Science and Engineering has been accepted as partially satisfying the requirements for "Java Project."

Approval of :

Supervisor : . . . . . .

Monisha Dey, Department of Computer Science and Engineering

Mymensingh Engineering College

Abstract

The aim of this project is to develop our skills in the Java language and make a sotfware that will help us to understand how to create a software. LiveScores provides free computer software capable of recording and analyzing local, regional, domestic, and recreational cricket matches.This simple, basic, and potent tool, which compliments the LiveScorers app, is completely customisable, making it the perfect choice for the dedicated scorer.

Contents

Chapter 1

1.1 Introduction

This collection of information or computer commands is known as software. Programs and data that are run on a computer are referred to as software. When referring to a piece of software, the word program is used.

In addition, software refers to all sorts of programs that are utilized to run computers and other electronic devices, such as smartphones and tablets. When it comes to computers, software is considered the changeable component whereas hardware is considered the invariable component. There are two types of software: application software (programs that users are directly interested in) and software (which encompasses operating systems and any program that supports application-level applications). It's common to use the word "middleware" to describe software that serves as a bridge between two distinct forms of application software (for example, a remote work request from an application in one operating system to an application in another operating system).

Ada Lovelace penned a blueprint (or algorithm) for the Analytical Engine, a hypothetical first piece of software. She devised mathematical arguments to demonstrate how the engine would compute Bernoulli numbers. As a result of her proofs and technique, she has been dubbed the "first computer programmer".

It was Alan Turing's 1935 work On Computable Numbers, with an application to the Entscheidungsproblem, that suggested the first notion of software (decision problem).

"The Teaching of Concrete Mathematics" by John Wilder Tukey was identified to have used the phrase "software" in a search of JSTOR's electronic archives in 2000 by Fred Shapiro, a Yale Law School librarian, two years before the OED cited it. Several obituaries published in the year of Tukey's death stated that he had coined the word, however Tukey never claimed credit for it. Paul Niquette claimed in 1995 that he created the word in October 1953, but he couldn't uncover any evidence to back up his claim. It was Richard R. Carhart's Rand Corporation Research Memorandum from August 1953 that used the word "software" in an engineering context for the very first time.

Software can be divided into three categories:

1. Application software: Software that goes beyond the capabilities of a computer's built-in features to accomplish additional tasks or give enjoyment to the user. Because current computers can accomplish so many various functions, there are many distinct kinds of application software.

2. System software: This is software that controls the behavior of computer hardware in order to provide users with basic functionality or to allow other software to run correctly, assuming it does so at all. In addition to providing a platform for executing application software, system software is also developed for this purpose.

3. Malicious software or malware: This is software designed to hurt and disturb computers. Malware is hence unwanted. Malware is often connected with computer crimes, while some dangerous programs may have been created as a joke.

Software is often planned and generated (aka coded/written/programmed) on integrated development environments (IDEs) such as Eclipse, IntelliJ, and Microsoft Visual Studio, which simplify and compile the process. Software is often built on top of existing software and the application programming interface (API) provided by the underlying software, such as GTK+, JavaBeans, or Swing. The purpose of libraries (APIs) may be classified. Data structures like hash tables, arrays, and binary trees, as well as techniques like quicksort, may be beneficial while developing software.

Software is frequently designed and created (also known as coded/written/programmed) using integrated development environments (IDEs) like Eclipse, IntelliJ, and Microsoft Visual Studio, which simplify and compile the process. The application programming interface (API) supplied by the underlying software, such as GTK+, JavaBeans, or Swing, is often created on top of existing software. Libraries (APIs) may be categorised based on their function. While developing software, data structures such as hash tables, arrays, and binary trees, as well as algorithms such as quicksort, may be useful.

1.2 Background Studies

Java is a high-level, object-oriented programming language, class-based with a low number of implementation dependencies. It is a general-purpose programming language designed to allow programmers to write once and run everywhere (WORA), which means that generated Java code may run on any systems that accept Java without the need for recompilation. Java programs are often compiled to bytecode that can run on any Java virtual machine (JVM), independent of computer architecture. Java's syntax is comparable to those of C and C++, although it has fewer low-level features than any of them. The Java runtime offers dynamic features (such as reflection and runtime code change) that conventional compiled languages do not. Java was one of the most popular programming languages in 2022.

1.3 Objectives:

Our system has multiple interfaces for interacting with the administrator and end users that can be used to manage any cricket match properly. The landing page appears upon startup, with several options for the user or administrator. These are to create a new player, to create a new team with 11 players, to set up a match between two teams, to see the results of a specific match or to watch live updates, and last but not least, an administrator update section. The first window is the player creation window, which contains all of the necessary input fields for obtaining the properties of a new player. When a new player is created, our system adds the player to our database. The team creation window then appears, allowing the administrator to create a team of 11 players, and a new team is added to the database. The same is true for arranging a match between the teams. People can view live match results by entering the match id, which redirects to the relevant matches. Only an administrator can update a match by providing the secret passkey, ensuring dependability and preventing unauthorized access.

1.4 Developing the System:

Our course is called "Object Oriented Programming." Object-oriented programming is the process of generating objects that have both data and methods.

There are various benefits to object-oriented programming versus procedural programming.:

- OOP gives the programs a clear structure.
- OOP aids in keeping Java code DRY (Don't Repeat Yourself), making it easier to maintain, modify, and debug.
- OOP allows you to create fully reusable applications with less code and in less time.
- OOP is quicker and simpler to implement.

Java is a class-based object-oriented programming (OOP) language built around the concept of objects. OOP concepts are intended to improve code readability and re-usability by defining how to structure your Java program efficiently. The core principles of object-oriented programming are:

Abstraction, Encapsulation, Inheritance, Aggregation,Composition, Polymorphism.

Each Object-Oriented Programming (OOP) idea in Java has its own code structure. For example, the extends keyword is used for the inheritance idea, while the greater and setter methods are used for the encapsulation idea.

Even though these ideas are important for making well-structured Java applications, adding crash reporting may also help you find problems that your end users are having during the maintenance and use phase of the software creation life cycle.

There are certain stages involved in software development.:

i. Problem analysis

ii. Market analysis

iii. Obtaining specifications for the proposed business solution

iv. Creating a strategy or design for a software-based solution

v. Software implementation (coding)

vi. Software evaluation

vii. Implementation

viii. Maintenance and bug fixes These stages are commonly referred to as the SDLC, or software development lifecycle.

Chapter 2

Implementation

The construction of the proposed system with the specified design in a procedural manner is referred to as implementation. This section of the report will discuss the System's implementation and how we did it.

To build our system, we decided to use JavaFX, a popular and modern Java GUI framework. JavaFX employs an MVC architecture, with interfaces created using markup languages such as XML (JavaFX has its own XML implementation that is FXML). In the MVC architecture, these interfaces are referred to as views and can be used to interact with the user. A controller class, which is a Java class, can be used to manipulate interfaces. These classes are in charge of routing and event handling. Any GUI-based system relies heavily on event handling. When a specific event occurs, it is handled by the appropriate event handler methods in the respective controller classes. Finally, in the MVC architecture, there is the model. A model is simply a class in the MVC architecture that is used to shape the data that the system works with. These classes provide the developer with a complete visualization of the data that needs to be integrated with. In our software, these elements combine to form a complete MVC pattern.

Chapter 3

Result and Discussion

3.1 Introduction

The main goal of this software is to create Cricket Match Scoring system software. We completed our project successfully after completing all of the procedures. All of the project's input and output are operational. As a result, our goal of running the program was met.

3.2 Experimental Output

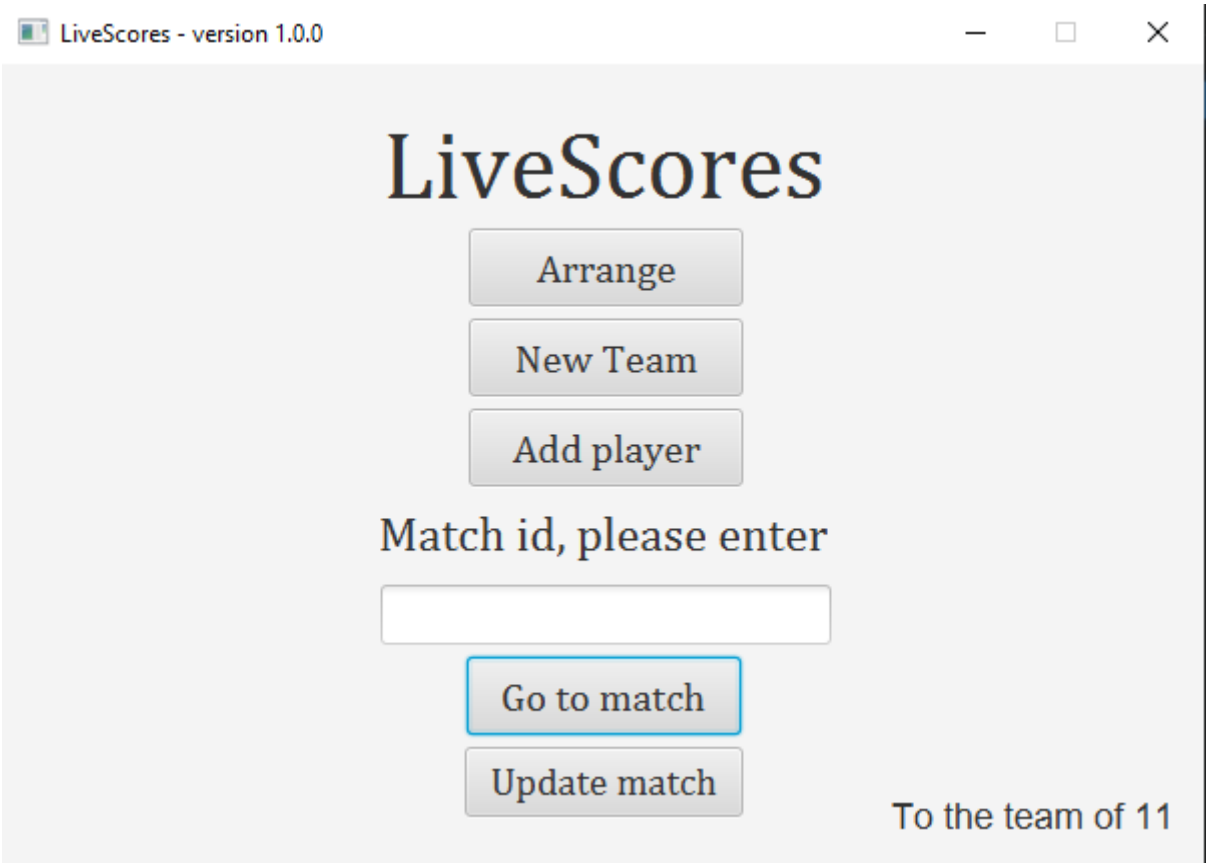

Figure 3.1: Homepage of LiveScores Software

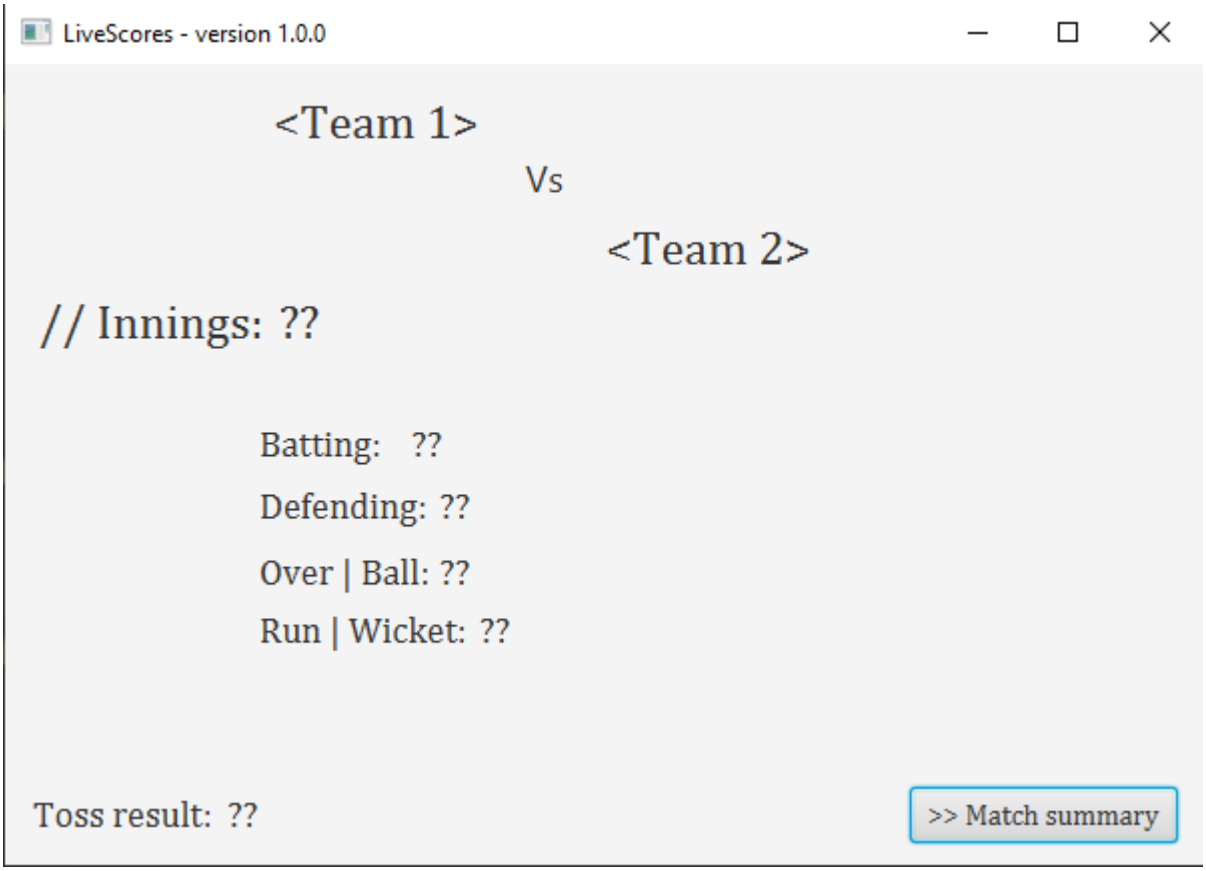

Figure 3.2: Result page of LiveScore Software

3.3 Limitation of this Project

1. We have not provided the API for our system due to time limitations.
2. The user interface at current stage is barely user friendly that needs a lot of enhancement.
3. Our database resides on local machines as no cloud services used leading to lack of reliability and severe chances of data loss.

3.4 Conclusion

The main goal of this software is to create Cricket Match Scoring system software. We completed our project successfully after completing all of the procedures. All of the project's input and output are operational. As a result, our goal of running the program was met. This is a corporate project as well. Softsasi LLC will publish this software once it has met corporate software standards and will provide a freemium bundle. As a result, it may also be financially beneficial to us.