Isaac Venable

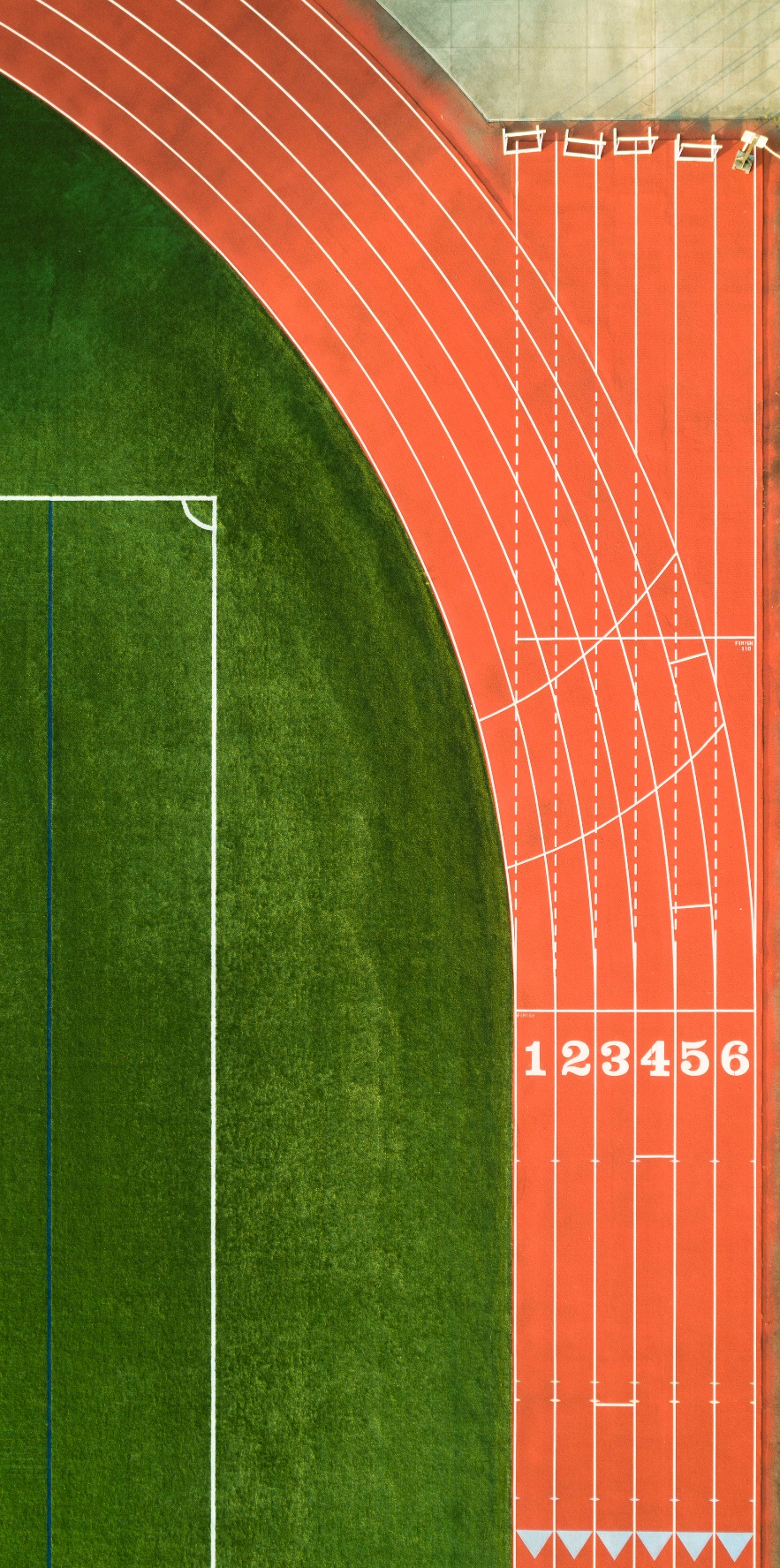
CSC3150 Systems Design

System Specification

6/2/2024

Professor Andy Cameron

**SmartSplits**

****

Contents

[1. Executive Summary 2](#_Toc168519566)

[2. Introduction 2](#_Toc168519567)

[3. Structural Model 3](#_Toc168519568)

[4. Architecture Design 5](#_Toc168519569)

[5. User-Interface 6](#_Toc168519570)

[6. Appendices 12](#_Toc168519571)

1. Executive Summary

SmartSplits is an app that gets splits and makes calculations for track and field athletes during their races. This system specification discusses the system in deep enough depth for a development team to start working. The customers for this app will be both track and field athletes and coaches, who will be able to purchase the app either individually or through a team plan. Development on the app has not yet begun, but with the completion of the document, development can start.

This document starts with an introduction that will give a basic overview of what the project is about. The rest of the document focuses more on the actual structure of the project.

1. Introduction
   1. **Problem Statement / Project Vision**

The customers for SmartSplits will be individual track and field athletes and coaches. A major element of track and field long-distance racing is pacing. Often, runners will not have access to splits, which can negatively impact a race, as a runner might accidentally run too slow or fast during certain parts of the race. Even if a runner has access to basic splits from either a clock or somebody timing them, you still cannot fully utilize that information without performing calculations, which can be difficult to do mentally, especially when one should be focused on the race. In addition to the need for splits during a race, runners can learn a lot about their performance by analyzing their splits after a race, but most of the time, race organizers don’t record splits.

We have an opportunity to build an app that will provide valuable information to track athletes and help them improve and become the best runners they can be. SmartSplits will help current track athletes with the motivation to keep training and give new runners the confidence to try out competitive racing. This could cause track and field to become more popular and motivate people to stay in shape.

* 1. **System Capabilities**

01. Record Splits – The app can record splits at different intervals for track races of any length.

02. On Pace For – The app determines what a runner is on pace to run, based on either average or most recent split.

03. Splits Needed – The app determines what a runner needs to run for each split based on previous splits and a goal time.

04. Split Comparison – The app can determine how much a runner is ahead or behind their previous races at any point in a race.

05. Save Splits – The app stores all races and splits in a database the user can access and analyze.

06. Pace Notifications – The app has different sound effects based on whether the runner is on pace.

(See section 4 of the System Proposal for more details)

* 1. **Non-Functional Requirements and Design Constraints**

Small development team – We plan to have only two developers for this app, but the project size means this should be easily completable despite the small team size.

Low budget – We do not have a lot of funding for this project, but we are maximizing what we do have by having a small team and not requiring many external resources.

Dependence on audience – Our app will only succeed if it becomes popular with our target audience, which is not something we can guarantee.

(See sections 1 and 4 of the System Proposal for more details)

* 1. **System Evolution**

We have plans for future versions of the app after the MVP (Minimum Viable Project) is developed. Adding a text-to-speech feature as an upgrade to the notification system would be a high priority for the first update. Updates past that would include the capability to track multiple runners at once or to add a workout suggestion generator.

* 1. **Document Outline**

The rest of this document will go in-depth into every area of the SmartSplits app, starting with the structural model, then the architecture design, then the user interface, and finally, the appendices.

1. Structural Model
   1. **Model Introduction**

This section will contain both the class diagrams and associated metadata with each class that will be needed for the app.

* 1. **Class Diagrams + Metadata**

|  |
| --- |
| ActiveRace |
| -raceName: String = <None>  -runnerName: String = <None>  -lapsTotal: Integer = <None>  -lapLength: Integer = 400  -lapsComplete: Integer = <None>  -goalTime: Integer = <None>  -startTime: Integer = <None>  -splitTimes: Integer[] = <None> |
| +Split()  +UndoSplit()  +SkipSplit()  +CancelRace()  +SaveRace() |

ActiveRace Metadata:

Description: Represents the race that is currently being timed

Visibility: Public

Is Abstract: No

**Attributes:**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Description | Read-Only | Multiplicity |
| raceName | The name of the race | No | 1 |
| runnerName | The name of the person racing | No | 1 |
| lapsTotal | The number of splits the race will take | No | 1 |
| lapLength | The length in meters of each lap | No | 1 |
| lapsComplete | How far along in the race the runner is | No | 1 |
| goalTime | A goal time chosen by the user to compare splits against | No | 1 |
| startTime | The exact time the race started | No | 1 |
| splitTimes | An array containing all of the splits from the race | No | 1 |

**Operations:**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Description | Is-Query? | Is-Polymorphic? |
| Split | Splits to the next lap, saving the time into splitTimes and updating all the race calculations | No | No |
| UndoSplit | Deletes the last split and undoes calculations (in case of accidental split) | No | No |
| SkipSplit | Moves forward past a split without logging a time (in case of missing the split) | No | No |
| CancelRace | Stops and deletes the race | No | No |
| SaveRace | Turns the ActiveRace into a CompletedRace which gets uploaded to the database | No | No |

|  |
| --- |
| CompletedRace |
| -raceID: Integer = <None>  -raceName: String = <None>  -runnerName: String = <None>  -lapsTotal: Integer = <None>  -lapLength: Integer = 400  -splitTimes: Integer[] = <None> |
| +DeleteRace()  +GetRaceName()  +GetRunnerName()  +GetTime()  +GetDistance() |

CompletedRace Metadata:

Description: Represents a race that has already finished and is in the database

Visibility: Public

Is Abstract: No

**Attributes:**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Description | Read-Only | Multiplicity |
| raceID | Unique identifier for the race | Yes | 1 |
| raceName | The name of the race | Yes | 1 |
| runnerName | The name of the person who raced | Yes | 1 |
| lapsTotal | The number of splits the race had | Yes | 1 |
| lapLength | The length in meters of each lap | Yes | 1 |
| splitTimes | An array containing all of the splits from the race | Yes | 1 |

**Operations:**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Description | Is-Query? | Is-Polymorphic? |
| DeleteRace | Deletes the race from the database | No | No |
| GetRaceName | Returns race name | Yes | No |
| GetRunnerName | Returns runner name | Yes | No |
| GetTime | Returns final race time | Yes | No |
| GetDistance | Returns race distance | Yes | No |

1. Architecture Design
   1. **Architecture Overview**

SmartSplits has a fairly simple architecture setup, as it doesn’t need to interact with any server or outside database. All of the UI, processing, and database storage will be done locally on the phone.

This section will include some diagrams describing the architecture, the hardware and software requirements, and a security overview.

* 1. **Infrastructure Model**
     1. **Deployment Diagram – Architecture Overview**

A screen shot of a phone

Description automatically generated

* + 1. **Deployment Diagram – Nodes and Artifacts**

A screen shot of a computer

Description automatically generated

* 1. **Hardware and Software Requirements**

The description describes the hardware and software needed to develop the MVP.

* + 1. **Hardware Components**

The app does not require any servers to run. Macs will be needed to develop the app. SmartSplits users will need to have an iOS mobile device to be able to use the app.

* + 1. **Required Software Components**

The app will be developed with XCode on Macs, and will be designed to run on iOS. The database will be created using SQLite through CoreData. To create all system design documents, Microsoft Word and LucidChart will be used.

* 1. **Security Plan**
     1. **Security Overview**

SmartSplits doesn’t interact with the internet or any servers in any way, and it also doesn’t store any sensitive personal information. This means that the security risks for the app are almost negligible. The main security concerns are during the development of the app.

* + 1. **Security Plan**

|  |  |  |
| --- | --- | --- |
| Threat | Physical (Computer damaged/stolen) | File (Corrupted/accidental delete) |
| Development Macs | 1,2 | 1 |

1. Create backups of all code
2. Require passwords to access code
3. User-Interface
   1. **User-Interface Requirements and Constraints**

This section covers how the app's UI is going to look, including how the screens connect with each other, how each screen will look, and how any output from the app will look. The app will be kept to a very limited amount of screens to allow for easy navigation and no wasted time for the user. The big UI issue with our app is the actual person running the race will not be able to see the screens during their race, so they will either have to rely on the app's sound effects or another person to communicate the necessary information.

* 1. **Window/Screen Navigation Diagram**

This diagram shows the basic navigation between the different screens of the SmartSplits MVP.

A diagram of a computer

Description automatically generated

* 1. **UI Wireframes**

This section will give a basic overview of how each screen of the app is going to look. Some screens will have additional notes.

* + 1. **Home Screen**

A screenshot of a phone

Description automatically generated

* + 1. **Race Setup**

The cancel race button will have an additional pop-up to confirm cancellation.

A screenshot of a phone

Description automatically generated

* + 1. **Active Race**

The save race button will only appear once the race is finished. The cancel race button will have an additional pop-up to confirm cancellation.

A screenshot of a phone

Description automatically generated

* + 1. **Completed Race**

The delete race button will have an additional pop-up to confirm the deletion.

A screenshot of a phone

Description automatically generated

* + 1. **Completed Races List**

A screen shot of a phone

Description automatically generated

* 1. **Reports: "Formal Output" Design**

For the MVP, SmartSplits will not have any formal output. In the future, we could add a feature to export data in an Excel-compatible format.

1. Appendices
   1. **Glossary**

Split: In track and field, split is the term for a runner to run a small part of their race, usually one lap

XCode: A popular IDE for MacBooks and other Apple products

MVP: Minimum Viable Product

UI: User Interface

* 1. **References / Bibliography**

Cameron, A. Various CSC 3150 Lectures. Seattle Pacific University, Seattle, WA.

* 1. **Supporting Documentation**

N/A