Spec – Smart Marathon running app

|  Submission date: 25-10-2024Student Name: Luka BrennanStudent Code: C00272285Supervisor: Chris Meudec  |
| --- |

[**1. Introduction 2**](#_s2zvowpa233y)

[**3. Application of the App 4**](#_74qpt5tt2m7f)

[**4. Possible Risks 4**](#_elztlm4pqcs)

[**5. Core functionates 4**](#_nkcgiebjbyxk)

[5.1 Integration 5](#_5brrjhuoemro)

[5.1 Menstrual cycle calculation 5](#_4qtlsr6hp9kz)

[5.3 Setback handling 5](#_v3c162ey8ev4)

[5.4 Feedback 5](#_2aezxmp7hdpj)

[5.5 Weather insight 5](#_pcvtuqj4g832)

[**6. Target users 6**](#_x731bg7z5ad6)

[**7. Use Case 6**](#_93pwl3o07khb)

[**8. Brief on Use Cases 6**](#_pefiwjpgvwl5)

[**Use Case: CRUD Account
Actor: Runner, Strava API
Description: 6**](#_18u5u8q2bwyn)

[**Use Case: Login
Actor: Runner
Description: 7**](#_gliz1w4whwqs)

[**Use Case: CRUD Plan
Actor: Runner, Strava API, Open Weather API
Description: 7**](#_5i4mf0jhu089)

[**Use Case: Get Feedback
Actor: Runner, Strava API, Open Weather API
Description: 7**](#_i6y1ga5fgszl)

[**8.1 Detailed use case 7**](#_528bckusvw0b)

[**9. Non Functional Requirements 8**](#_4fywvxsj1nk3)

[9.1 Performance 8](#_7kjkg4asq1ps)

[9.2 Multiple users 8](#_421khldiiw0z)

[9.3 Availability 8](#_nqx7pk4a7cjq)

[9.4 Usability 9](#_m66ppx6nr9ya)

[9.5 Security 9](#_sy1g5298y5df)

[**10. Requirements for 1st iteration 9**](#_idfgv7snyh9t)

[**11. Target Platform 9**](#_c45bzbolc5t1)

[**12. Metrics 10**](#_27uw0f1k80em)

[**13. Similar Apps 10**](#_a57u3vc7l9y9)

[**14.References 14**](#_hy7ieusguvu0)

# **1.** **Introduction**

Many marathon runners encounter obstacles during their training, whether due to medical issues, injuries, or life circumstances. These setbacks can cause them to lose valuable training time, negatively affecting their performance. Current training plans often fail to adapt to such challenges, leading runners to push themselves prematurely, risking further injury.

This application aims to address that gap by providing a smart, adaptive marathon training app. The app dynamically adjusts the runner’s training plan based on real-time data, such as heart rate, pace, and gender-specific factors like the menstrual cycle. It also accounts for external factors like weather conditions, which can impact a runner's performance. By integrating with APIs such as Strava and Open Weather, the app continuously refines the runner’s plan, ensuring they can safely and effectively improve, even in the face of setbacks.

**2. Project overview**

Marathon training and running is a popular fitness people enjoy doing, however many training programs and apps are too static or generic. This could for instance mean not having dynamic changes be made to a plan. The main aim for this application is to be a non static marathon training app, allowing dynamic changes be made without the need for runner’s input. The app will make use of Strava’s API, this will allow runner’s who use Strava to use Smart Marathon Running App as a means of getting the most out of their training.

|  |  |  |
| --- | --- | --- |

**Figure** No. 1 - Stravas different uses [5]

# **3.** **Application of the App**

**Figure** No.2 - Business Model Canvas

This app will provide runners with a running plan that dynamically adapts to their running patterns. As many other marathon train apps are static in the sense that they don’t change automatically if the runner may have not run in a while. The vision for this running app is to allow runners to create their running program and if any unfortunate occurrence happens to the runner, the app will acknowledge this and make necessary changes to ensure that the runner will be able to get back to training efficiently.

# **4. Possible Risks**

* Difficult with integration with Strava API and Open Weather API. These could be rate limits or service outages.
* The app may incorrectly detect a setback, this could be falsely identifying if a runner has an injury or period related performance issues.
* Relaying on real time weather data could be wrong or might not always work if there are delays.

# **5.** **Core functionates**

##  **5.1 Integration**

The app will allow users with different watches to use the app, as long as they are compatible with Strava. The app will use metrics from Strava to determine the best pace increase for the runner on their next run. The app also takes into account a runner's gender, if a runner is a woman then changes would be made during that runner’s menstrual cycle to make their run’s more achievable.

##  **5.1 Menstrual cycle calculation**

The app will allow female runners to input key details when creating their account, these details would be the start date of their last period and an average cycle length. With this data the app will be able to adapt and change the plan accordingly. There are three phases to this the **Follicular phase** [6] (Day 1 - 13), which will make the runner feel better and would allow them to perform better. **Ovulation phase**[6] (Day 14), during this time hormone release including estrogen and testosterone are at its highest, this would be the best time for the runner to train and the app would increase the pace for runs during this time. **Luteal phase** [6] Day(15 - 28), During this time fatigue and other physical symptoms will arise causing training to be difficult, due to this the app will lower the pace needed for runs during this time to still allow for training just not as hard.

## **5.3 Setback handling**

If a runner misses a scheduled run or has sustained an injury, the app will adjust for this ensuring a smooth recovery to get the runner back to training. This will be done by comparing their pace and heart rate, if they have drastically changed over the span of a day the app will make the decision that the runner has sustained an injury and feedback message will be provided once they have completed their run for confirmation. If they select no then the app will not make drastic changes to their required pace, else the app will reduce the pace required to ensure that the runner is not doing more damage to their body.

## **5.4 Feedback**

The app will notify runners with helpful feedback after they have completed a run. Depending on the run if it either was an increase or decrease the feedback message will provide useful information on how to improve.

## **5.5 Weather insight**

As many runners train in often harsh environments such as hot humid summer days or cold rainy days, the app will take these different weather conditions into account when performing calculations on either increasing or decreasing the runner’s pace, as well as feedback on performing in these conditions. For example if the weather for a certain day is 23 degrees celsius with a 75% humidity this would drastically increase the runner’s heart rate, so using this data, calculations would be made to get a good estimate as to either increase or decrease pace. Using Open Weather API [11] will be crucial in the predictions of weather conditions that could alter the runners performance.

# **6.** **Target users**

As this app uses data gathered from Strava users who want to use this app need to have a Strava account set up and have a compatible smart watch with them (either FitBit, Garmin or Apple watch). So, the app is mainly aimed towards Strava users who want a more concise training plan to use for improvement in marathons.

# **7.** **Use Case**



**Figure** No.3 - Use Case for the Smart Marathon Running App.

# **8.** **Brief on Use Cases**

# **Use Case: CRUD AccountActor: Runner, Strava APIDescription:**

The runner opens the app for the first time and is prompted to enter their personal information, such as name, gender, and other relevant details. After filling in all the fields, the runner links their Strava account to the app. Once the process is complete, they are taken to the main screen to begin using the app.

# **Use Case: LoginActor: RunnerDescription:**

The runner opens the app and logs in using their credentials. After logging in once, the app remembers their session, so the runner can access the app in the future without needing to log in again.

# **Use Case: CRUD PlanActor: Runner, Strava API, Open Weather APIDescription:**

When the runner logs into the app, they are prompted to input their target race distance (marathon for now) and the number of weeks remaining until the race. Based on this information, the app generates a personalized training plan using established methodologies like **Pfitzinger & Douglas** [10]. The plan adapts dynamically, adjusting to the runner's performance using Strava data, weather conditions via the OpenWeather API, and, if applicable, menstrual cycle information or injury history. After each run, the runner’s plan updates to refine pace for future runs.

# **Use Case: Get FeedbackActor: Runner, Strava API, Open Weather APIDescription:**

Upon completing a run, a feedback message will be displayed to the runner. This feedback message will showcase what they completed in their run, giving statistics such as pace per kilometer, elevation, heart rate and cadence. The feedback message will include the necessary adjustments for the next run based on the data taken from this completed run, such as a pace increase or decrease.

# **8.1 Detailed use case**

**Name: Get Feedback**

**Actors: Runner, Strava API, Open weather API**

**Description:**

This use case begins once a runner has completed a run. The feedback message will include the runner's accomplishments for that run, this will include the average pace per kilometer, heart rate and weather they trained during harsh weather conditions. If the runner was a female and was training during their Luteal phase of the menstrual cycle the app will provide information on whether they have pushed themselves too much. The feedback message may also say that “pace has increased for the next run” or “Pace has decreased for next run” depending on the data taken from that run. Finally There will be a “See Training plan” button at the bottom of the feedback message directing the screen to the CRUD plan screen.

**Main Success Scenario:**

1. Runner completes their run.
2. The app acknowledges this and provides a feedback message for that run.
3. Runner clicks the “See training plan” button.
4. Runner is brought back to the training plan.

**Alternatives:**

 2.a: Runner has no internet access.

 The app will not be able to provide the necessary feedback.

 1.a: The app does not detect a finished run.

# **9. Non Functional Requirements**

## **9.1 Performance**

* **Main Goal**: The app should be able to respond to the runner’s data for a completed run and decide to either increase or decrease their pace.
* **Quantified Requirement**: The app will have to process this data to give an accurate increase or decrease in pace for their plan within 2 - 5 seconds of receiving data from Strava. Feedback should be delivered within 2 seconds after processing.

## **9.2 Multiple users**

* **Main Goal**: The app should be able to handle multiple users without giving false or incorrect feedback and info.
* **Quantified Requirement**: The app should be able to support at least 100 concurrent users during the first iteration.

## **9.3 Availability**

* **Main Goa**l: The app requires consistent access to the internet in order to make necessary API calls.
* **Quantified Requirement**: API calls to Strva should be successful 99% of the time while being connected to the internet.

## **9.4 Usability**

* **Main Goal**: The app should provide a user friendly interface that is easy to navigate.
* **Quantified Requirement**: 90% of runner’s should be able to navigate the app’s UI to access their training plan without the need for assistance. The screens for the app should load within 0.5 - 1 seconds, depending on the runner's internet connection.

## **9.5 Security**

* **Main Goal**: The app should not provide incorrect information to different runners, this would include giving one runner another runners info.
* **Quantified Requirement**: The app should ensure a 100% success rate in preventing data leaks.

# **10.** **Requirements for 1st iteration**

The primary objective for the first iteration is to implement core features that enable the app to gather data from runners and their Strava account, process that data to deliver meaningful pace adjustments. The main use cases that will be developed during this iteration include:

**Training Plan (CRUD Plan)**

The aim is to provide the runner with a plan which shows their current completed run and their previous completed run. The app will then gauge whether or not to increase the pace, decrease the pace or keep the same pace, based on the two runs. This will then provide a new plan with a new pace to aim for. All this will be determined by the heart rate, pace per kilometer and vo2 max [7].

# **11.** **Target Platform**

As Android is the most widely used mobile operating system in the world (71.65%) [1]. The version of android that the app will be able to work on is android 11.0 up to android 14.0 [2].

# **12.** **Metrics**

The success of the application will be determined by whether it meets the following criteria.

* **Core functionality**: The app will have a functioning structure that allows for user account creation, logging in and accessing a personal adaptive training plan.
* **Strava Integration**: The app should successfully connect to Strva, allowing the app access to the runner’s data, such as heart rate, distance, pace and other metrics from their previous runs.
* **Adaptive Training**: The app must dynamically adjust the training plan based on the runner’s previous runs which include their pace and setbacks.
* **User Feedback**: The app should be capable of delivering personalized feedback messages to the runner, giving them tips and info based on today's run.
* **Setback Management**: The app should be able to detect potential setbacks and make necessary adjustments to the training plan accordingly, without the use of runner input.

# **13.** **Similar Apps**

There are a few popular apps that share similar functionalities to what is being made such as Strava and Garmin Connect. However, these apps have limitations when it comes to dynamic training plans and handling setbacks.

Strava [3], Is one of the most popular fitness tracking apps out there, being widely used by average runners and marathon runners. It allows users to track their activities, analyze performance metrics, and share their progress with a community of athletes. Strava syncs with several fitness devices and can track routes, pace, elevation, and heart rate. However Strava’s training plan is static and requires a subscription to gain access to a better version of the plan.

**Figure** No.4 - Strava paid subscription[8]



**Figure** No.5 - Strava paid features[8]

Strava also does not dynamically adjust workouts based on health data such as VO2 max, menstrual cycle, or injury setbacks.

Garmin Coach [4], offers personalized training plans for runners through Garmin Connect. It’s tailored toward users who own Garmin devices and provides adaptive training plans for beginners to intermediate runners. Thus this being made for only Garmin watches means that a large portion of runners don't get access to the app. That being said, runners who have Garmin watches gain access to tailored training plans that are based on the runners goals. With the plan adapting based on runner performance. Garmin coach is perfect and free for Garmin watch runners and provides the fundamentals for what this app aims for [9]. The only criticism that can be made is the lack of compatibility being locked to only Garmin watch runners, as well as not having full customisation, like incorporating features such as injuries, menstrual cycle or weather conditions. These could all influence a runner's training and having these features would improve training.



**Figure** No. 6 - Garmin Coach run overview [9]

# **14.References**

[1] Android Statistic as of sep 24 2024 by Ahmed Sherif -<https://www.statista.com/statistics/272698/global-market-share-held-by-mobile-operating-systems-since-2009/#:~:text=Android%20maintained%20its%20position%20as,percent%20during%20the%20same%20period> [Accessed on 26-09-2024].

[2] Popular android versions as of September 2024 by StatCounter -<https://gs.statcounter.com/android-version-market-share> [Accessed on 26-09-2024].

[3] Strava app - [https://www.strava.com](https://www.strava.com/) [Accessed on 27-09-2024].

[4] Garmin Coach -<https://connect.garmin.com/features/coach/> [Accessed on 27-09-2024].

[5] Different screens of Strava by Mark Dredge - <https://therunningchannel.com/strava-updates-2024-night-heatmaps-ai-dark-mode/> [Accessed on 21/10/2024].

[6] The Impact of Menstrual Cycle Phase on Athletes’ Performance: A Narrative Review by Mikaeli Anne Carmichael, Rebecca Louise Thomson, Lisa Jane Moran and Thomas Philip Wycherley as of February 2021 - <https://pmc.ncbi.nlm.nih.gov/articles/PMC7916245/#:~:text=Physical%20performance%20has%20been%20postulated,may%20affect%20muscle%20strength%20and> [Accessed on 24/10/2024].

[7] How to use Vo2 mam edited by Fabi Fuu 76 on 1st October 2024 - <https://en.wikipedia.org/wiki/VO2_max> [Accessed on 23/10/2024].

[8] Strava’s Paid Subscription - <https://www.strava.com/subscribe?code=training-plan-upsell&origin=training_plan> [Accessed on 23/10/2024].

[9] Garmin coach review by Sherry Fijas - <https://wrinkledrunner.com/garmin-coach-review/> [Accessed on 23/10/2024].

[10] Advanced marathoning by Pete Pfitzinger and Scott Dougals - <https://books.google.ie/books?hl=en&lr=&id=rz6lDwAAQBAJ&oi=fnd&pg=PR3&dq=advanced+marathoning+&ots=xm_WlfURNT&sig=GIjRaiakZEd9sytC_rBsdnp8t8E&redir_esc=y#v=onepage&q=advanced%20marathoning&f=false> [Accessed on 24/10/2024].

[11] Open Weather API - <https://openweathermap.org/api> [Accessed on 24/10/2024].