FootyBettor

Sports Betting Platform (SBP)

Business Requirements Document

Prepared by:
Zahra Cheeseman
Esa Chen
Junhao Qu
Shamdeed Kabir

Release Date:
Wednesday 1st March, 2023

Contents

Executive Summary

Sports Betting Platform (SBP) is a unique form of entertainment that combines the excitement of sports with the thrill of gambling. SBP is intended to exist in similar markets as other sports betting websites, but provides more data-backed and informed odd calculations, and uncluttered, understandable and easy to use operation pages. SBP softwares will be distributed as a standalone application for use on end user hardware without network connectivity.

This document provides nontechnical information regarding the purpose and behaviour of SBP.

Document Versioning

Date	Owner	Comment
03/01/2023	Zahra Cheeseman, Esa Chen, Shamdeed Kabir, Junhao Qu	Features, Feature Matrix, and Feature Discussion
03/01/2023	Zahra Cheeseman	Executive Summary
03/01/2023	Zahra Cheeseman	Project Description
03/01/2023	Esa Chen	Use Case 1 and Use Case 2
03/02/2023	Zahra Cheeseman	User Stories
03/02/2023	Shamdeed Kabir	Use Case 4-7
03/02/2023	Shamdeed Kabir	Added Features
03/02/2023	Shamdeed Kabir	Added Feature Discussion
03/02/2023	Esa Chen	Use Case 3
03/02/2023	Esa Chen	Added User Stories
03/02/2023	Junhao Qu	Added sketch diagrams
03/02/2023	Esa Chen	Table of Contents

Project Description

FootyBettor is looking to enter the sports gambling market. This market is saturated with platforms with clunky and complex operation pages, limited data availability and implicit probability distribution. Sports Betting Platform (SBP) is FootyBettor's creation which provides a solution to these problems which traditional betting platforms present which create difficulty in users being able to make informed decisions, place winning bets and succeed in the highly competitive sports betting industry. SBP allows for impressive data visualisation of previous season team statistics and a smooth user experience for easy betting on the result of matches between sports teams. Users can create an account in which they navigate through the program, and view a balance which they can add to and which gets updated in relation to the return on their bets. SBP generates precise odds which are informed by a multitude of statistics used in calculations which optimise accuracy.

SBP is highly extendable and thus would have potential to be able to support different applications such as a variation in the sport selected and type of bet available.

Additionally, SBP can support a graphical user interface, however does not necessarily require one to interact with the program. The program may be run completely via command line interface support. Currently, there does not exist a consummate sports gambling program which is for low power and low capability embedded platforms. SBP provides a resolution which supports such devices plus ones which have sufficient requirements to support a GUI, allowing for an increased potential pool of devices.

Features

The feature matrix enumerates the features requested for the project and the discussion section provides details regarding the intent of the feature. The ids will be used for traceability. Features that all stakeholders have agreed can be removed should strike-through the feature id and have a comment added to discuss the feature being dropped.

Priority Codes:

H - High, a must have feature for the product to be viable and must be present for launch
M - Medium, a strongly desirable feature but product could launch without
L - Low, a feature that could be dropped if needed

Feature Matrix

ID	Pri	Feature Name	Owner	Comment	Case #
s.1	H	Multiplatform	Sales		
s.2	M	No coding required	Sales		
s.3	H	Standalone application	Sales		
s.4	L	IP Protection	Sales		
ux.1	H	CLI support	Design		
ux.2	L	GUI support	Design		
ux.3	H	Error messages	Design		
e.1	M	Error reporting	Support		
e.2	H	Menus	Eng		
e.3	H	Fixed pathway	Eng		
e.4	H	User	Eng		
e.5	M	Tracking balance	Eng		
e.6	M	Stats Presentation	Design		
e.7	H	Odds calculation	Eng		
e.8	H	Return calculation	Eng		
e.9	M	Flexible keywords and digits	Eng		
e.10	M	Cash in/Cash out Balance	Eng		

Feature Discussion

S.1 - Multiplatform

To support marketing and expand potential market penetration, SBP must be able to run on the most common computer platforms. MacOS, Windows, and Linux must be supported.

S.2 - No coding required

Studios adopting SBP need not have staff developers to use the betting application. Coding is out of scope for the users of the SBP and such the program must be implementable as data file(s) read by SBP.

S.3 - Standalone application

SBP is envisioned as a standalone platform that itself has a multitude of functionality which can satisfy all of the user's requirements. There does not need to be any additional subscription or micro transactional revenue models, as Footybettor does not support these exploitative practices.

S.4 - IP Protection

SBP and all the intellectual property should not be modified without authorization. Mitigations against easily modifiable features should be included in SBP.

UX.1 - CLI support

Text interaction is not as computationally complex as GUI support and does not require real time interaction. However, command line interface support can still provide a similar, exciting game experience which can also increase the potential pool of devices SBP can run on.

UX.2 - GUI support

Most modern computer users are unfamiliar with the command line and expect a GUI which is easy to use, and displays options clearly and aesthetically.

UX.3 - Error messages

When SBP encounters any issues that make the platform unable to run/load (i.e unable to handle user input, picking games, loading data), an error message of suitable format should be displayed and informed to the user.

E.1 - Error reporting

When SBP encounters any issues while running the program (i.e. loading user data, calculating values, not being able to handle user input etc.), it will create an error report with enough detail so that developers can debug the code and fix it.

E.2- Menus

There would be multiple executables that a user can choose from a menu bar such as start betting, adding money to the account, displaying chart.

E.3 - Fixed pathway

There is a certain, predetermined pathway of frames in the program which users can navigate through. This forms a fixed pathway in which users can only access specific locations based on their decisions, and can always navigate to the dashboard.

E.4 - User

SBP has a login interface at the first menu, and it requires a username and password to approve the users to access their accounts. It also provides a sign up button to let new users create their accounts.

E.5 - Tracking balance

SBP records the balance for each account, including the money from recharge and the money they earned from betting.

E.6 - Stats Presentation

SBP would have an option for presenting all of the teams' previous stats so that the user can get detailed information about the teams.

E.7 - Odds Calculation

SBP would show the odds of winning the bet on a certain team by calculation based on predetermined formulae/equations.

E.8 - Return Calculation

SBP would use the game results to determine whether the bettor won or lost their bet. Afterward, it will calculate the return on the bet based on the odds shown previously.

E.9 - Flexible keywords and digits

The system will provide case insensitive search results to users. Additionally, the system will convert any numerical value (i.e. decimal, integer, etc) to a standard format for compatibility of calculation.

E.10 - Cash in/Cash out Balance

The system would let the user to cash in or cash out the money available in their balance. Applicable situations could be: the user loses money and wants to add to their balance, or the user gains money and wants to cash out their full or a portion of their balance.

User Stories

The primary users of SBP are bettors. SBP provides an encompassing platform for a wide variety of bettors, including users who are new to betting or frequent bettors, and sports fanatic bettors or users less familiar with sports. SBP provides resources and an uncomplicated user interface which allows accessibility for a range of people.

Mark, a football fanatic

Mark is a 35 year old football fanatic. He keeps close tabs on all of the games occurring in the premier league, and can recite any historic statistic on all of the players in every club. He wants to use SBP to earn as much money as possible, as he has memorised the results of all the previous games, and thinks he can predict the outcome of any upcoming game.

Dustin, a frequent bettor

Dustin is a 65 year old frequent bettor. He has spent many years of his life betting on horses, on slot machines, and at casinos. He has never betted on sports before, so SBP provides an exciting new avenue for him to try.

Samantha, new to betting

Samantha is 30 years old, and has never placed a bet before. She likes to use SBP because she finds the platform very easy to navigate and simple to use. She can view all of the previous game statistics, which helps her decide how much to wager and who to bet on.

To reliably support all these users, a careful consideration of how SBP users might use the platform and how SBP can create a user-friendly, accessible, straightforward interface to provide a better using experience for the users is required.

The following use cases describe goals, criteria for success, and potential extensions for failures:

Use Case 1

Name	Start/Pick a game
ID	UC01
Description	The user wants to pick a game to bet on
Actor	User
Organisational Benefits	FootyBettor wants many users to pick games to bet on as much as possible
Use Frequency	Very frequent
Triggers	User makes a choice and picks a game by clicking one on the window

Preconditions	The window with the list of all games has pumped out
Main Success Scenario	• User picks a game • System closes current window
Extensions (Error Scenarios)	• User doesn't pick a game • System continue on the same window
Alternative Courses	• User selects back to home • System closes current window and redirects user to the dashboard
Post Conditions	System calculated odds of the two teams in the selected game with a new window including a text field for the user to enter wager

Use Case 2

Name	Bet money
ID	UC02
Description	The user wants to bet money on the selected game
Actor	User
Organisational Benefits	FootyBettor wants users to bet more money on selected games
Use Frequency	Very frequent
Triggers	User enters the wager in the text field on the window
Preconditions	The window which shows calculated data allowing users to make an informed decision about the result, and which includes a text field for users to enter wager, has popped up. User has chosen the winning team
Main Success Scenario	• User enters wager • System checks the user's balance • System saves wager the user entered
Extensions (Error Scenarios)	• User enters wager larger than current balance • System closes current window and pops up a new one with an error message of not enough money in balance and an option of adding money into balance • System present the window allowing user to input amount of money to add to balance • System redirects to dashboard
Alternative Courses	• User selects back • System closes current window and redirects to the window which allows the user to re-choose a game
Post Conditions	System closes current window and presents a new window with the actual result and amount of money to be returned to user's balance

Use Case 3

Name	Choose the winning team
ID	UC03
Description	The users wants to choose which team of the two will be the winner
Actor	User
Organisational Benefits	FootyBettor wants users to interact with the platform and perform actions
Use Frequency	Very frequent
Triggers	User enters the chosen winning team in the text field
Preconditions	The window which shows calculated data allowing users to make an informed decision about the result, and which includes a text field for users to enter wager, has popped up
Main Success Scenario	• User enters one of the two teams of the game in the text field as the predicted winner • System verifies the chosen team • System stays on the current window
Extensions (Error Scenarios)	• User enters a blank team name or an incorrect team name of the selected game • System shows an error message of wrong input and waits for a new input from user
Alternative Courses	• User selects back • System closes current window and redirects to the window which allows the user to re-choose a game
Post Conditions	System stays on the current window and waits for the user input for wager

Use Case 4

Name	Ask for team stats
ID	UC04
Description	The user wants to view the stats of the teams' previous performances
Actor	User
Organisational Benefits	FootyBettor wants user to feel educated about the teams' performances and feel confident when they bet by providing them with extensive data
Use Frequency	Varied; based on user behaviour
Triggers	User clicks view team stats option in the dashboard

Preconditions	The dashboard has successfully presented the option of viewing team stats
Main Success Scenario	● User successfully clicks the team stats button on the dashboard ● System is able to load the team stats data without error ● System successfully displays the team stats window
Extensions (Error Scenarios)	● User clicks a wrong option ● System fails to load the data
Alternative Courses	N/A (since the user is on dashboard)
Post Conditions	System keeps displaying the team stats window until the user clicks the return option.

Use Case 5

Name	Change Balance
ID	UC05
Description	The user wants to add money or take out money from their balance sheet
Actor	User
Organisational Benefits	FootyBettor wants to provide the users with the flexibility of adding in or taking out money from their balance sheet anytime and any number of times
Use Frequency	Varied; based on user success/behaviour
Triggers	User clicks the account balance option on the dashboard
Preconditions	The dashboard successfully loads the window and loads the account balance amount without error.
Main Success Scenario	● User successfully clicks the account balance option ● The system successfully loads the amount of available balance from the database ● System successfully transitions to account balance window
Extensions (Error Scenarios)	● User picks wrong input ● System fails to access data for the available balance
Alternative Courses	N/A (since the user is on dashboard)
Post Conditions	The system shows the new balance after calculating the change asked by the user

Use Case 6

Name	Win bet
ID	UC06
Description	The user wins the bet by correctly predicting the game result
Actor	User
Organisational Benefits	FootyBettor wants to return the money to users if they win the bet
Use Frequency	Frequent
Triggers	User clicks the bet option, which will confirm their betting information. Afterward, the system will display the result of the game and show the amount of money won
Preconditions	The window asking for betting information for the game has received correct input
Main Success Scenario	• User input correct amount of money to bet on based on their account balance • User successfully selects the result of the game to bet on • System successfully processes the user input • System successfully displays the next window show the result of the bet
Extensions (Error Scenarios)	• User provide wrong input for the amount to bet on • System fails to retrieve data form the database to calculate odds
Alternative Courses	• User selects back • System closes current window and offers user to pick a game again
Post Conditions	The system shows the user won of the bet, money won, the changed account balance of the user

Use Case 7

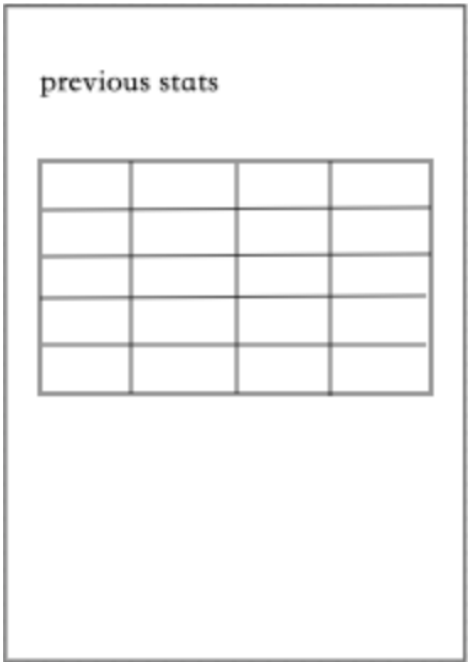
Name	Lose bet
ID	UC07
Description	The user loses the bet by incorrectly predicting the game result
Actor	User
Organisational Benefits	FootyBettor wants to gain money every time a user loses their bet
Use Frequency	Frequent

Triggers	User clicks the bet option, which will confirm their betting information. Afterward, the system will display the result of the bet and show the amount of money lost
Preconditions	The window asking for betting information for the game has received correct input
Main Success Scenario	• User input correct amount of money to bet on based on their account balance • User successfully selects the result of the game to bet on • System successfully processes the user input • System successfully displays the next window show the result of the bet
Extensions (Error Scenarios)	• User provide wrong input for the amount to bet on • System fails to retrieve data form the database to calculate odds
Alternative Courses	• User selects back • System closes current window and offers user to pick a game again
Post Conditions	The system shows the user has lost the bet, money lost, the changed account balance of the user

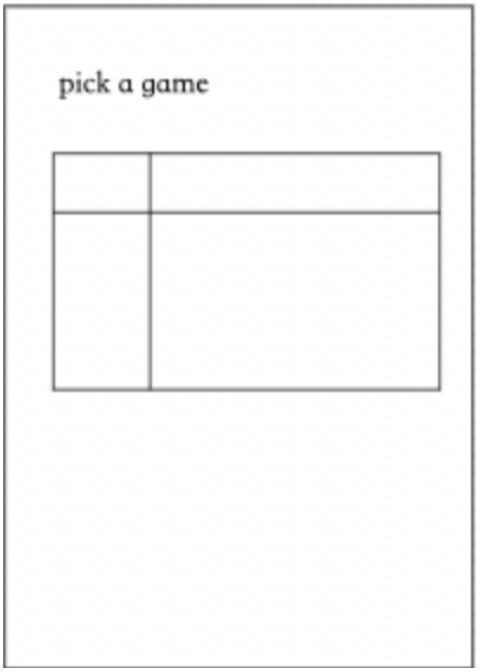

login
username:
password:
sign up


create new account
username:
password:


dashboard
team stats
start betting
recharge
current balance: XXX


previous stats


recharge
current balance: XXX
add money:


pick a game


odds
win: XXX
lose: XXX
draw: XXX
wager:
current balance: XXX


No enough money in balance
recharge


result: XXX
current balance: XXX
dashboard