

Masters Degree in Informatics Engineering
Internship
Final report

Traderline - Mobile application development to support sports betting

Bruno Miguel Morais Caceiro
caceiro@student.dei.uc.pt

Advisor from DEI-FCTUC:
Prof. Dr. Pedro Abreu

Advisor from *IPN-LIS*:
Eng. Alcides Marques

Coimbra, 1st July, 2016



FCTUC DEPARTAMENTO
DE ENGENHARIA INFORMÁTICA
FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Masters Degree in Informatics Engineering
Internship
Final report

Bruno Miguel Morais Caceiro
caceiro@student.dei.uc.pt

Advisor from DEI-FCTUC:
Prof. Dr. Pedro Abreu

Advisor from *IPN-LIS*:
Eng. Alcides Marques

Judges:
Prof. Dr. Álvaro Rocha
Prof. Dr. Carlos Fonseca

Coimbra, 1st July, 2016



FCTUC DEPARTAMENTO
DE ENGENHARIA INFORMÁTICA
FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Abstract

Betting exchanges pose as a new way of making bets, here bettors bet against each other, without the intermediaries such as bookmakers. Betfair is a betting exchange which specialises in sports betting. The focus of this project is the development a mobile application to support sports trading on this betting exchange, named Traderline.

Various tools exist which allow the user to execute a betting strategy against live market data, trading softwares. These tools rely on the performance and data updates, since the Betfair website is slow compared to professional trading software. There are some products available in the market but Traderline poses as a complete and tailor-made package for trading, with advanced interfaces, statistic centre, livenesscores and betting strategies.

Stratio, owner of Traderline, needs to develop a complete new application that supports all operating systems, both mobile and desktop, thus the technical solution:Xamarin, which allows to share code across multiple projects.

Developing a Mobile application, integrated in a cross-platform solution allows a great amount of code shared, accelerating the development process while maintaining a reliable application. The first semester focused in learning the fundamentals of trading, the new Technology in which the application would be developed and implementation of a determined number of requirements. In the second semester all the remaining requirements were implemented, the application was tested and validated and a version of Traderline was released to the Asian Market, having already more than 1000 people around the world using the application, with 251 active users, over than 22.3 million euros exchanged and more than 100 thousand transactions.

In the future, it would be interesting to add a recommender system to provide alerts to the user. This approach could also be generalised or transformed into a gamification concept, promoting the continuous interaction of the user with the application.

Resumo

As bolsas de apostas são uma nova forma de fazer apostas, onde apostadores apostam uns contra os outros, sem os intermediários denominados como *bookmakers*. A Betfair é uma bolsa de apostas que se especializa em apostas desportivas. O principal objectivo deste projecto é o desenvolvimento de uma aplicação móvel, chamada Traderline, que possibilite a compra e venda de apostas para esta bolsa em específico.

Existem várias ferramentas diferentes para levar o utilizador a optar pela melhor estratégia de apostas possível fazendo uso da informação existente no mercado, o chamado software de apostas. Estas ferramentas tentam distinguir-se através da performance e taxa de actualização de informação, uma vez que o website da Betfair é, comparativamente a software de apostas profissional, bastante mais lento. Apesar de existirem alguns produtos disponíveis no mercado, o Traderline apresenta-se como um pacote completo e personalizado para apostas, com interfaces avançados, centro de estatísticas, resultados em tempo real e estratégias de apostas.

A Stratio, empresa responsável pelo Traderline, precisa de desenvolver uma aplicação de raiz que suporte todos os sistemas operativos, tanto mobile como desktop. Daí fazer-se uso da seguinte solução técnica: Xamarin, que permite a partilha de código através de múltiplos projectos.

Desenvolver uma aplicação móvel, integrada numa solução de múltiplas plataformas permite partilhar uma grande quantidade de código, acelerando o processo de desenvolvimento enquanto se mantém a sua fiabilidade. O primeiro semestre focou-se na aprendizagem dos fundamentos das apostas, da nova tecnologia a ser utilizada para o desenvolvimento da aplicação e da implementação de um determinado número de requisitos. Durante o segundo semestre todos os restantes requisitos foram implementados, a aplicação foi testada e validada e lançada no mercado Asiático. Tendo já sido utilizada por mais de 1000 pessoas, 251 utilizadores activos, e realizadas mais de cem mil transacções envolvendo, aproximadamente, 22.3 milhões de Euros.

No futuro, seria de particular interesse fazer uso de um sistema de recomendações, de modo a fornecer alertas ao utilizador. Esta abordagem poderia ser também generalizada ou ainda transformada num conceito de gamificação, de modo a promover uma interacção contínua do utilizador com a aplicação.

Acknowledgments

I deeply appreciate my advisors Eng. Alcides Marques and Prof. Dr. Pedro Abreu, both for their motivation, guidance, insights and patience.

My appreciation to the team behind IPN-LIS, for all the support during the internship.

A special thanks to José Pedro Marques, for being a true friend and support while battling through all the adversities. To my friends and colleagues, in a unique way each and everyone was important. And finally to my parents, who gave me this opportunity and always wanted the best for me.

Contents

1	Introduction	3
1.1	Motivation	4
1.2	Objectives	5
1.3	Methodology and Working plan	6
1.4	Planning	7
1.5	Document Structure	10
2	Business Knowledge	11
2.1	Trading	11
2.1.1	Odd	11
2.1.2	Exchange	12
2.1.3	Events	13
2.1.4	Back	13
2.1.5	Lay	13
2.2	Betfair	14
3	Competitors Analysis	17
3.1	BetOnDroid	17
3.2	Betfair Mobile Application	18
3.3	Conclusions	19
4	Technical Approach	23
4.1	Xamarin	23
4.1.1	Hardware Features	24
4.1.2	Code Sharing	24
4.1.3	Native Bindings	25
4.1.4	Xamarin Forms	25
4.2	Conclusions	26
5	Requirements Analysis	27
5.1	Functional Requirements	28
5.1.1	Requirements for Login	28
5.1.2	Requirements for Favourite Sports	28
5.1.3	Requirements for General Options	29
5.1.4	Requirements for All Sports Page	29
5.1.5	Requirements for Search Page	29
5.1.6	Requirements for Main Page	30
5.1.7	Requirements for Market Interface	30
5.1.8	Requirements for Grid View	31

5.1.9	Requirements for Bet Settings Page	32
5.1.10	Requirements for Registration page	32
5.1.11	Requirements for Update System	32
5.2	Non-Functional Requirements	33
5.2.1	Security	33
5.2.2	Extensibility	33
5.2.3	Usability	33
5.2.4	Performance	34
6	Solution	35
6.1	Technologies	35
6.2	Architecture	36
6.2.1	Level 0	36
6.2.2	Level 1	37
6.3	Project Structure	38
6.4	Results	40
6.4.1	Implemented Requirements	41
6.4.2	Traderline Application progression	47
6.5	Testing	50
6.5.1	Scripted Testing	50
6.5.2	Exploratory Testing	50
7	Conclusions	53
A	Flow Diagrams	59
A.1	Login	59
A.2	Home	60
A.3	Settings	61
A.4	Search	62
A.5	Live	63
A.6	Favourite Sports	64
A.7	Favourite Sports - Football	65
A.8	All Sports	66
A.9	General Settings	67
A.10	License	68
A.11	Main Sports	69
A.12	My Markets	70
A.13	Market	71
A.14	My Bets	72
A.15	Grid	73
A.16	Bet Slip	74
A.17	Bet Settings	75
A.18	Other Markets	76
A.19	Stake Options	77
A.20	Ladder	78
A.21	Ladder Runners	79
A.22	Multibet	80
A.23	Stats Center	81
A.24	In Play	82

<i>CONTENTS</i>	xi
A.25 Charts	83
A.26 Standings	84
A.27 Live Score	85
B Test Scripts	87

List of Tables

1.1	Traderline Team	4
3.1	Comparison table between different Betfair registered Betting Exchange applications.	20
5.1	Requirements specification for Login Page	28
5.2	Requirements specification for Favourite Sports Page	28
5.3	Requirements specification for General Options Page	29
5.4	Requirements Specification for All Sports Page	29
5.5	Requirements Specification for Search Page	29
5.6	Requirements Specification for Account View	30
5.7	Requirements Specification for Live View	30
5.8	Requirements Specification for Market	30
5.9	Requirements Specification for Stakes	30
5.10	Requirements Specification for Grid View	31
5.11	Requirements Specification for My Bets View	31
5.12	Requirements Specification for Statistics Page	31
5.13	Requirements Specification for Ladder View	31
5.14	Requirements Specification for Multibet View	32
5.15	Requirements Specification for Bet Settings Page	32
5.16	Requirements Specification for Registration Page	32
5.17	Requirements Specification for Update System	32
5.18	Non-Functional Requirement 1 - Security	33
5.19	Non-Functional Requirement 2 - Extensibility	33
5.20	Non-Functional Requirement 3 - Usability	34
5.21	Non-Functional Requirement 4 - Performance	34
6.1	Bugs detected in the previous Traderline Application	48
6.2	Traderline versions comparison	48
6.3	Scripted Test Case Scenario Example	50
B.1	TC-LOGIN-01: Login in the application	87
B.2	TC-LOGIN-02: Remember me	87
B.3	TC-LOGIN-03: Practice Mode	87
B.4	TC-LOGIN-04: Practice Mode	88
B.5	TC-LOGIN-05: Register Betfair account	88
B.6	TC-LOGIN-06: Retrieve account password	88
B.7	TC-FAVSPORT-01: Access Favourite Sports	88
B.8	TC-FAVSPORT-02: Filter Favourite Sport Events	88

B.9	TC-GENOPT-01: Australian Markets	88
B.10	TC-GENOPT-02: Market View Mode	89
B.11	TC-GENOPT-03: Cancel unmatched bets	89
B.12	TC-GENOPT-05: Hedge value in Ladder	89
B.13	TC-GENOPT-06: Ladder column order	89
B.14	TC-GENOPT-07: Inverted values in Ladder	90
B.15	TC-GENOPT-08: Inverted values in Ladder	90
B.16	TC-GENOPT-09: Live Results	90
B.17	TC-GENOPT-12: Account balance	90
B.18	TC-GENOPT-13: Application sounds	90
B.19	TC-GENOPT-14: Favourite Sports	91
B.20	TC-ALLSPORT-01	91
B.21	TC-SEARCH-01	91
B.22	TC-MARKET-01	92
B.23	TC-STAKE-01	92
B.24	TC-MYBETS-01	92

List of Figures

1.1	Traderline Mobile Software Development Process	6
1.2	First Semester Planning with deviations	8
1.3	Second Semester Planning with deviations	9
3.1	BetOnDroid application screenshots.	18
3.2	Betfair application screenshots.	19
4.1	Xamarin C# libraries bindings to native OS SDKs and .NET [1]	24
4.2	Xamarin Forms architecture [1]	25
4.3	Xamarin.Forms Pages represent cross-platform mobile app screens [2].	25
4.4	Xamarin.Forms Layouts [2]	26
6.1	Level 0 architecture diagram	36
6.2	Level 1 architecture diagram	37
6.3	Traderline Solution Structure	38
6.4	Traderline.Forms Project Structure	39
6.5	Traderline.Droid Project Structure	39
6.6	Traderline Mobile application (Phone and Tablet)	40
6.7	Login and Home Page Screenshots	41
6.8	All Sports and Search View Screenshots	42
6.9	Market Views (Grid, Ladder) Screenshots	43
6.10	Multibet and My Markets View Screenshots	44
6.11	Filter by Sport and Statistic Centre Screenshots	45
6.12	Menu View and Application Settings Screenshots	46
6.13	Previous Traderline Application and the New version	49
A.1	Login Page Diagram	59
A.2	Home Screen Page Diagram	60
A.3	Application Settings Page Diagram	61
A.4	Search Page Diagram	62
A.5	Live Page Diagram	63
A.6	Favourite Sports Page Diagram	64
A.7	Favourite Sports - Football Diagram	65
A.8	All Sports Diagram	66
A.9	General Settings Diagram	67
A.10	License Diagram	68
A.11	Main Sports Diagram	69
A.12	My Markets Diagram	70
A.13	Market Screen Diagram	71

A.14 My Bets Diagram	72
A.15 Grid View Diagram	73
A.16 Bet Slip view Diagram	74
A.17 Bet Settings Diagram	75
A.18 Other Markets View Diagram	76
A.19 Stake Options View Diagram	77
A.20 Ladder View Diagram	78
A.21 Ladder Runners Diagram	79
A.22 Multibet View Diagram	80
A.23 Stats Center Diagram	81
A.24 In-Play view Diagram	82
A.25 Charts view Diagram	83
A.26 Standings view Diagram	84
A.27 Live Score view Diagram	85

Acronyms

API Application Programming Interface.

IoT Internet of Things.

IPN Instituto Pedro Nunes.

LIS Laboratory for Informatics and Systems.

MIE Masters in Informatics Engineering.

SDK Software Development Kit.

XAML Extensible Application Markup Language.

1

Introduction

The present report reflects the work developed by the student Bruno Miguel Morais Caiceiro, regarding the discipline "Dissertation/Internship" from the Masters in Informatics Engineering (MIE), Faculty of Science and Technologies from University of Coimbra, in the academic year 2015/1016.

This internship is the result of a partnership between the Laboratory for Informatics and Systems (LIS) of Instituto Pedro Nunes (IPN) and Stratio [3], the product owner.

The Laboratory for Informatics and Systems (LIS) brings together a wide set of competencies and expertise in several areas of Information and Communication Technologies[4] (ICT) which are applied and used to support initiatives for technology transfer towards industry and public bodies.

Stratio develops software for mobile and desktop applications, as well as Internet of Things (IoT) solutions. Their partnership in the mobile and desktop environment is Betfair. The company also develops products regarding Intelligent Transportation Systems[5] and Industry 4.0[6].

Traderline is the result of the partnership between Stratio and Betfair (BET:LSE), a widely recognised software developed for sports trading. It is an essential tool used by several sports traders from the advanced and professional market segment, since it provides a similar approach to stock exchange, adapted to sporting events. Traderline is one of the principal products of Stratio, used by thousands of people, present in more than 50 countries.

Despite already having a multi-platform solution Traderline, Stratio was obliged, due to internal problems, to develop a completely new version of the software, without having any access to the previous codebase, making this the opportunity to refactor many aspects of the former applications. The previous Traderline applications (Desktop and Mobile), were built independently from each other, leading to a laborious and painful process to solve problems in each one. The old mobile application had several problems regarding performance, functionalities, design and, to sum up, quality. Being such an important product for the company, it was important to develop the new Traderline solution in a new technology, adopting practises and tools that would facilitate the development and maintenance of the product.

The intern is integrated in a development team that is developing the Traderline Solution (Windows, Mac, Mobile).

Table 1.1 illustrates the different roles and number of people involved in the Traderline Solution.

Responsability	Number of Elements
Traderline Mobile	1 (Intern)
Traderline Windows	3
Traderline Core Module	3
Traderline Mac	1
Designers	2
Stakeholder	1

Table 1.1: Traderline Team

By being integrated in this development team, the intern actively participates in the definition and development of the Traderline Solution. The communication between all team members is an additional effort, to keep the project synchronized. A constant communication is required within the different members, specially with the stakeholders, to gather information regarding the Traderline Mobile application, the requirements, the validation of the product. The main interaction occurs between the team behind the development of the Traderline Core Module, since it is one of the principal modules of the Traderline Solution, where the majority of the business logic is implemented and it is necessary to understand all the components, as well as having an active role during its development. Regarding the designers, there were several meetings to define the final version of Traderline.

1.1 Motivation

Gambling is a part of human nature, predicting the outcome of certain events has a special interest to many people. Adding to this, wagering money on such events has become a sport, an addiction. It is a growing industry, exchanging billions of dollars. With the development of technology, it is easier to bet and to win, or lose, money, but the tendency is to improve, to innovate, and mobile betting gives a chance to bettors to bet anywhere at anytime, without being constrained to a computer, at home.

Betfair is the big player of the betting exchanges, with its platform it attracted several gamblers to bet on sports, politics and many other events.

A mobile application for trading is the next step, it gives an edge for a bettor to be in the field, watching the event live and place his bets when compared to the traders betting at their homes, watching the event over a stream, with the corresponding delay. Traderline mobile provides all the necessary tools in the hands of amateur and professional traders. Advanced trading tools such as Dutching and Bookmaking are available for the first time on mobile devices, which poses as a revolution in the mobile betting industry.

Mastering the business beneath Sports Trading is a very interesting task, though requiring a long learning process. In a software perspective, the amount of data involved in sports trading poses as a challenge, by delivering an application that can maintain a high performance while still displaying frequently updated information.

Mobile development was always a personal preference of the intern, and being able to develop an android application, to sports traders is a great motivation, while being an opportunity to learn, to discover and understand Trading.

By being integrated in a development team, offers an opportunity to learn by the experience of others. Taking part in design and solution of a cross-platform application is

a completely new experience which opens new horizons regarding Software Engineering.

1.2 Objectives

Traderline is a cross-platform solution to sports trading and the main goal is to develop a new mobile solution based on a totally new common codebase that will be the source of data for the different presentation modules. To accomplish this, the new solution will make use of a relatively new technology: Xamarin.

The mobile application should take advantage of the models and business logic, regarding trading, developed in the core module by the Traderline development team, and display the data accordingly.

The goal of this Internship is to develop an Android application for Betfair users, that provides the betting and trading tools available at any time and place.

It will be necessary to implement all the application design as well as all the user interface logic, there is no code reuse from the previous version, since it is developed in a new technology and there is no access to the old codebase. The mobile application will be displaying frequently updated information and since performance is one of the non-functional requirements, knowledge about the Technology, and its optimisations will be required.

Thus, the goals of this Internship are as follows:

- Learn the Business behind Trading:
 - Understand the Betfair system, the process involving bets and Trades;
- Analysis and refinement of Requirements:
 - Understanding the previous developed application;
 - Dealing with client expectations:
- Master the Technology:
 - Understand the architecture of a cross-platform solution;
 - Learn how to deal with asynchronous communications, multi-threading, recurring tasks;
 - Data Bindings, events, collections;
 - Encrypting user information;
 - Best practices regarding performance, caching strategies;
 - Debugging android application;
- Integration in a software development team:
 - Communication to streamline the development process;
 - Learn from the experience of others;
- Testing:
 - Validate requirements implementation.

1.3 Methodology and Working plan

The software development process adopted in this internship is based on the Agile Principles. The Agile Manifesto focuses on individuals and interactions over processes and tools, giving emphasis to the working software rather than documentation [7]. The Agile methodology allows for changes to be made after the initial planning, overcoming possible changes that may be required in the development of the product. This means that during the development process, there is a complete set of tasks to be implemented, however if needed, the schedule may be rearranged according to the client's demand.

The Requirements were defined in a first phase of the project. During the development of the Traderline mobile solution, there was always the possibility of new requirements to appear, however it was not the most likely scenario. From the Requirements list, there was a list of tasks, which can relate to the Backlog in the Scrum Framework[8], that was regularly updated. In the development phase, the requirements would be implemented and then a working application would be sent to the client for validation. From the validation, issues might be raised, which would be added to the list of tasks and then in the development phase would be fixed for later validation. Figure 1.1 intends to reflect this process.

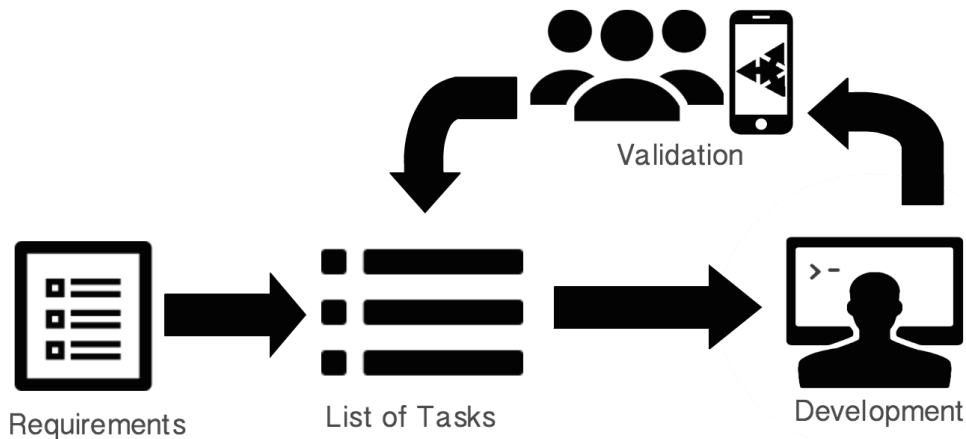


Figure 1.1: Traderline Mobile Software Development Process

In the context of this internship, the tasks to be performed were primarily established by the intern, according to the internship plan, and discussed with the project manager.

1.4 Planning

The activity plan for this internship was compiled with the project manager and prioritised according to the project needs while maintaining the integrity of the internship.

The tasks were mainly separated by the Application Screens to be implemented.

The first semester focused on learning the technology that would be used to develop the Traderline solution, following a learning phase regarding Trading. After this preparation phase, the development process was started. In order to develop each application screen and after an initial phase of requirements analysis it was necessary to implement the given design in the used Technology, the Models are developed in the shared code-base, a constant communication with the development team was essential to streamline the implementation. There was a phase in the semester, where a performance issue regarding the Traderline Mobile application was detected, due to the inexperience in the technology, which could only be detected after the application complexity increased, with the development of more screens. In order to correct the problem, there was a week dedicated to understand the causes of the low performance and fix the problem, which led to a deviation from the initial planning, postponing tasks. The Gantt chart (Figure 1.2) represents the planning for the first semester with the according deviations.

The second semester focused on development and validation, finishing all the defined requirements and validating the implemented features with the client. In the middle of this process, the application was deployed to a set of beta testers from the company, which resulted in a series of bugs and issues that were gathered and solved, before finishing the implementation of the remaining requirements. This implicated a deviation from the initial planning, postponing the development of the Multibet View to a later stage.

The chart in (Figure 1.3) represents the planning for the second semester.

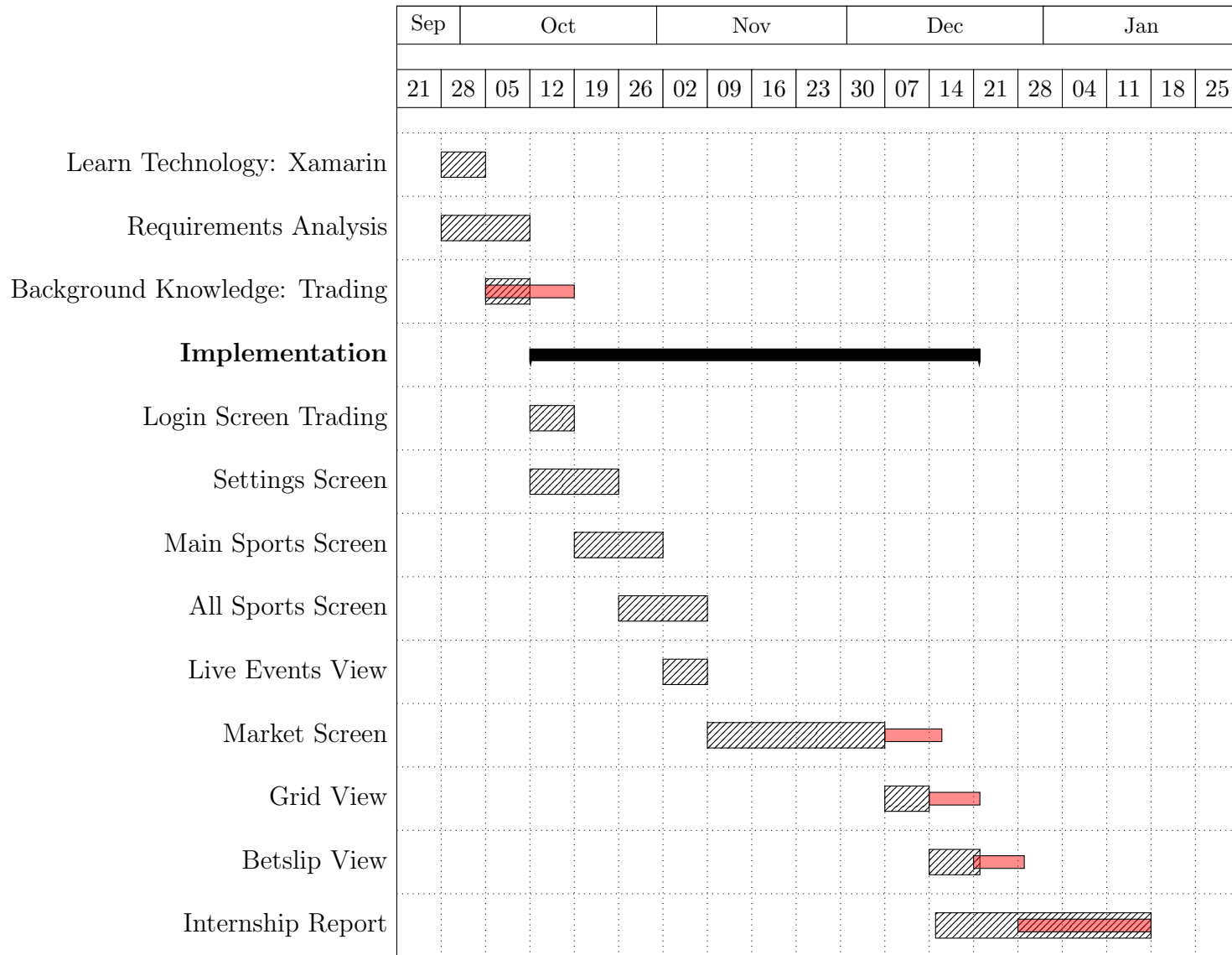


Figure 1.2: First Semester Planning with deviations

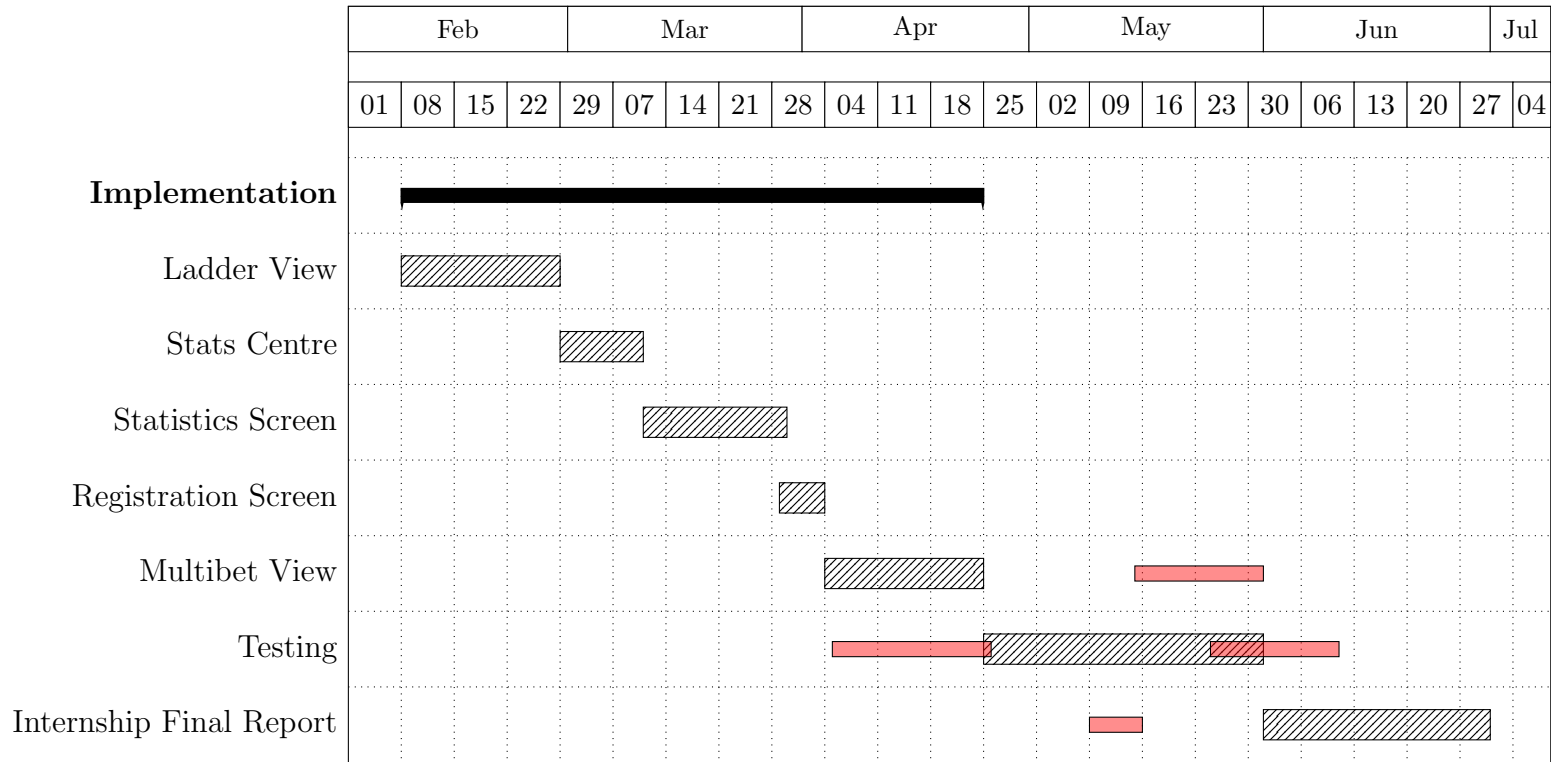


Figure 1.3: Second Semester Planning with deviations

1.5 Document Structure

Initially, the main concepts regarding Trading and Betfair will be presented within the Business Knowledge chapter. A competitors analysis will be detailed in the third chapter and chapter 4 will explain the Technical Approach for the project, respecting the Constraint imposed by the stakeholders. Later, in chapter 5, the Requirements Analysis will be detailed. The solution will be described in chapter six, specifying the systems architecture, the project structure and the implemented screens. followed by the Traderline Mobile validation.

To conclude, an analysis of the results of this internship as well some final thoughts and expectations for the future.

2

Business Knowledge

The main goal of this chapter is to explain the principal concepts regarding trading and, more specifically, sports betting in the betting exchange that is Betfair.

2.1 Trading

Trading involves the transfer of the ownership of goods or services from one person or entity to another in exchange for other goods or services or for money[9].

Sports betting is nowadays intrinsically related to trading. Here gamblers place bets on the outcomes of the different sporting events.

Traditionally, if a gambler wanted to make a bet on a certain outcome (such as the outcome of a football game), the bet would be placed with a bookmaker or with a parimutuel betting market. The bookmaker is an independent agent who decides the odds that he/she feels are appropriate for the estimated probability of the outcome and the customer can place bets against those odds. Bookmakers can expect to make profits in the long run by providing odds to their customers which slightly overestimate the true probability of a specific outcome[10].

A parimutuel betting market is a system in which all bets on the outcome of a particular event are placed together in a pool, and the payoff odds are calculated by sharing the pool among all winning bets[11].

In the context of stock market, trading refers to the process of buying, selling or exchanging auctions. In the sports context, it consists in betting in favour or against the probability of a certain event occurring. This probability is defined in the world of Trading as an **Odd** and it is possible to win money by the variation of odds, with bets against or in favour. The most important concepts in Trading are as follows: odd, back, lay. In order to explain these concepts, let us consider an example of a football match between two national teams: Benfica and Porto.

2.1.1 Odd

The odds on an event represent the probability of different outcomes and are used instead of percentages as they display the return on a wager[12]. There are three main styles of displaying odds: European (decimal odds), English (fractional odds) or American odds (moneyline).

$$Odd = \frac{1}{EventProbability}$$

Decimal Odds

Decimal odds, also known as European, represent the odd of an event occurring as the inverse of its probability[12].

$$DecimalOdd = \frac{1}{EventProbability}$$

In order to calculate the return value of a bet in decimal format we simply multiply the amount that we want to bet by the decimal odd value. Here the return value includes the initial stake.

Assuming that:

$$\begin{aligned} Odd &= 1.3 \\ Stake &= 100\text{€} \end{aligned}$$

Then:

$$Return = Odd * Stake = 130\text{€}$$

Fractional Odds

Fractional odds, also known as UK Odds [13] are most often used in the United Kingdom and Ireland. Originally used in horse racing, fractional odds are one of the oldest forms of odds for sports gamblers.

These odds show the amount that would be won excluding the wager as a ratio to the wager with both values being integers. They can be calculated as the ratio of the probability of an outcome not occurring to the probability of an outcome occurring[12].

$$FractionalOdd = \frac{(1 - EventProbability)}{EventProbability}$$

Lets consider the following fractional Odd:

$$4/1$$

Odds of 4:1 or "Four to One" would imply that for every 1€ staked, we would stand to win 4€. When the first number is higher than the second, it is called "odds against". If the first figure is smaller than the second, then it is called "odds on", and the amount of winnings will be smaller than the stake. Using the previous example, if the odds were to be 1:4 for every 4€ staked, we would just win 1€ [14].

2.1.2 Exchange

Betting exchanges are markets for traders to bet in the most different events. They differ from the traditional bookmakers since here, the gamblers bet against each other, not against the house (bookmakers), applying a fee on each transaction. Gamblers can bet against and in favour of a certain event, but the great advantage is the trading possibility. It is not necessary, for gamblers, to wait for a certain event to finish in order to win, or loose, money.

2.1.3 Events

In the betting exchanges it is possible to bet in a great amount of events, ranging from sports (football, tennis), or even politics and each event has different markets for the bettors to bet on. For example, in a football match, there is the Match Odds market, usually the market with most money involved. This market has different selections to bet on, these selections reflect the different possible outcomes in this specific market. If we consider a football match, Benfica vs Porto, the three possible selections are: Benfica to Win, Porto to Win or a Draw between the two teams.

2.1.4 Back

To **back** a selection in a market is to place a bet in favour of a certain selection to win. Using our example, Benfica vs Porto, we can consider three possible outcomes: the victory of Benfica, the victory of Porto, or the draw. For each possible outcome, we can place a back bet. We can Back Benfica for its victory, which means that we place a bet where Benfica will win the match.

The profit of a **back** bet is calculated using equation 2.1 and the liability (in case of loss) of a Back bet is the amount of the bet itself[15].

$$ProfitBack = Stake * (Odd - 1) \quad (2.1)$$

2.1.5 Lay

To **lay** a selection in a market is to place a bet against a certain selection, this means that the selection will not win. In the previous example, we backed Benfica for its victory, now let's consider we place the opposite bet, where we Lay Benfica for the victory, which means that we place a bet where Benfica will not win the match.

The liability or amount in case of loss of a Lay bet is given by equation 2.2 and the profit is the amount of the bet itself. Basically the Lay is the "mirror" of Back[15].

$$LiabilityLay = Stake * (Odd - 1) \quad (2.2)$$

As it was previously explained, betting exchanges offer the trading possibility. This consists of making different bets, back and lay, in order to maximise profit, or minimise the losses disregarding the final outcome of the event. This is where advanced tools like Traderline play an extremely important role, since they perform several calculations automatically, according to the different market (odds) variation.

A transaction must have the following elements:

- Plan of action. It is very important to anticipate the events and have a plan ready for any eventuality;
- Point of entry. It is possible to back or lay a team waiting for odds to low their price, or we can lay a team hoping that odds will increase their price;
- Point of exit. If we have backed a team at a certain price, the objective is to lay that selection at a lower price, in order to make a successful transaction (with profit). In the same line of thought, if we have layed a selection at a certain price, we must back the same one at a higher price;
- Share profit in all possible outcomes. This is the key for successful trading - the ability to profit regardless of the outcome.

2.2 Betfair

Betfair is a betting exchange, an online market for gamblers to trade in cultural and sports events all around the world. In these markets, gamblers bet against each other, not against Betfair, who generates revenue through its base commission of 5% in each trade. Betfair also provides a public API that allows companies to create products to automate bets, allowing trades with higher chances of success.

There are important concepts regarding Betfair that must be explained:

- **Hedging** - placing several back and lay bets in order to maximise the profits or minimise the losses disregarding the final outcome of the event.
- **Green up** - to hedge for a profit over all selections (green book);
- **Red Up** - to hedge for a loss on all selections (red book);
- **Scratch** - to cancel out a trade by placing the opposite bet at the same odds, or to cancel the bet before it is matched;
- **Stop loss** - bet placed (usually with the aid of trading software) to guarantee that the trader will not lose more than a specific amount on a given trade;
- **Tick** - the difference between the two closest prices at Betfair;
- **WOM (weight of money)** - reflects the market trends with the gamblers bets. For instance, if the gamblers are making more lay bets on a certain selection, the weight of money for lay is higher than for back.

Every market on the exchange (as example, the match Benfica vs Porto) consists of several outcomes with ex-ante objective probabilities of happening p_1, \dots, p_N . For every possible outcome of the event, bettors have two bet options:

- Back - place a bet that the outcome will happen;
- Lay - place a bet that the outcome will not happen.

The Betfair betting exchange is designed as an order-driven market where bettors can place either limit orders or market orders[16].

- **Market Orders** - bettor can only choose a side (back or lay), a particular outcome and a bet size. This bet will be later matched at the best possible price available on the market;
- **Limit Order** - bettor can choose everything, side, outcome, volume and also the odds at which he is willing to bet.

3

Competitors Analysis

There are few betting exchange mobile android applications in the market that are certified by Betfair[17]. In order to develop and distribute betting software to Betfair customers it is necessary to have a Software Vendor License, approved by Betfair. Currently there are three registered applications accessible in their Applications Directory: Traderline, BetOnDroid[18], Rebel Betting[19], and also the official Betfair Android application[20]. When accessing to the Rebel Betting website, they have no longer any reference to a mobile version of their software, and it isn't possible to find any application, so it was probably terminated. This concludes that, to this extent there are two direct competitors of Traderline Mobile: BetOnDroid and Betfair Mobile application. There are, however, several other applications for Desktop, but since they belong to a different scope, will not be considered for this analysis.

3.1 BetOnDroid

BetOnDroid is an application for betting and Trading at Betfair Exchange[18], available for the Android Platform[21]. The main features of this application are:

- Filter Events - easily find and filter events by sport, by championship, by date;
- Inplay Markets - Access to the markets that are going inplay, this means that they are live, or going live;
- Ladder Interface - A different view to analyse the market, with a list of all the odds available, the money exchanged in each odd;
- Multiple wallets - access both to Betfair UK markets and Australian Markets;
- Customization - Place bets with a simple click;
- Account - Manage betfair account and consult details.

Below in figure 3.1 it is possible to see some screenshots of the BetOnDroid application: the main screen, its Ladder View mode, and list of sports screen.

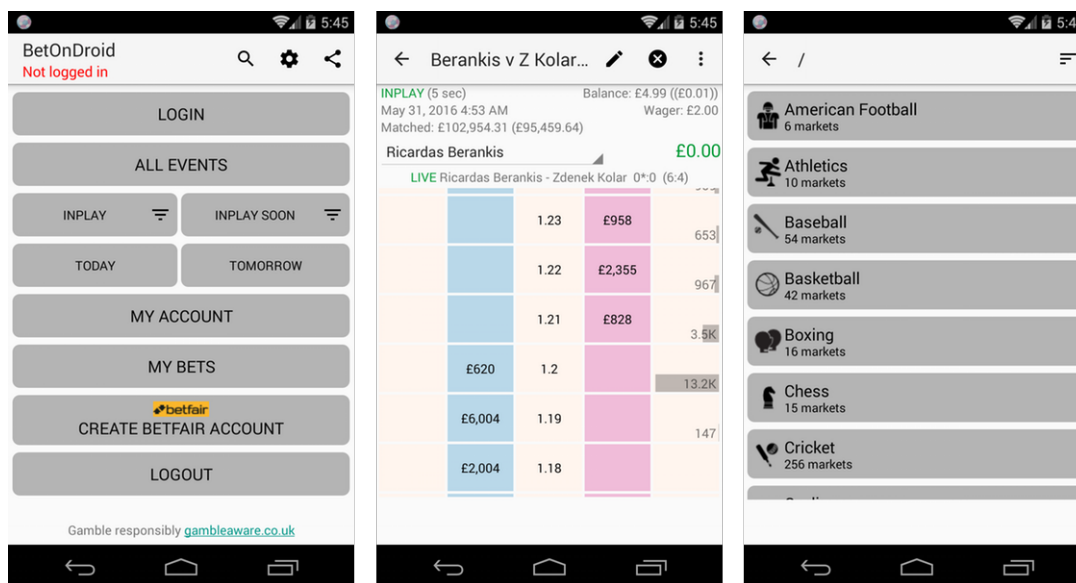


Figure 3.1: BetOnDroid application screenshots.

3.2 Betfair Mobile Application

Betfair has released an android application to complement its main platform (betfair.com) It is possible to access all the information available in the Betfair website, and this application features a live chat for users to easily get support.

The main features of this application are:

- Live In-Play betting;
- Best odds;
- Betting Exchange;
- Easy to use, fast and secure;
- Wide market choice;
- Live Video Streaming.

The interface of the Betfair Mobile Application is similar to their website as it is possible to see in the following screenshots (Figure 3.2).

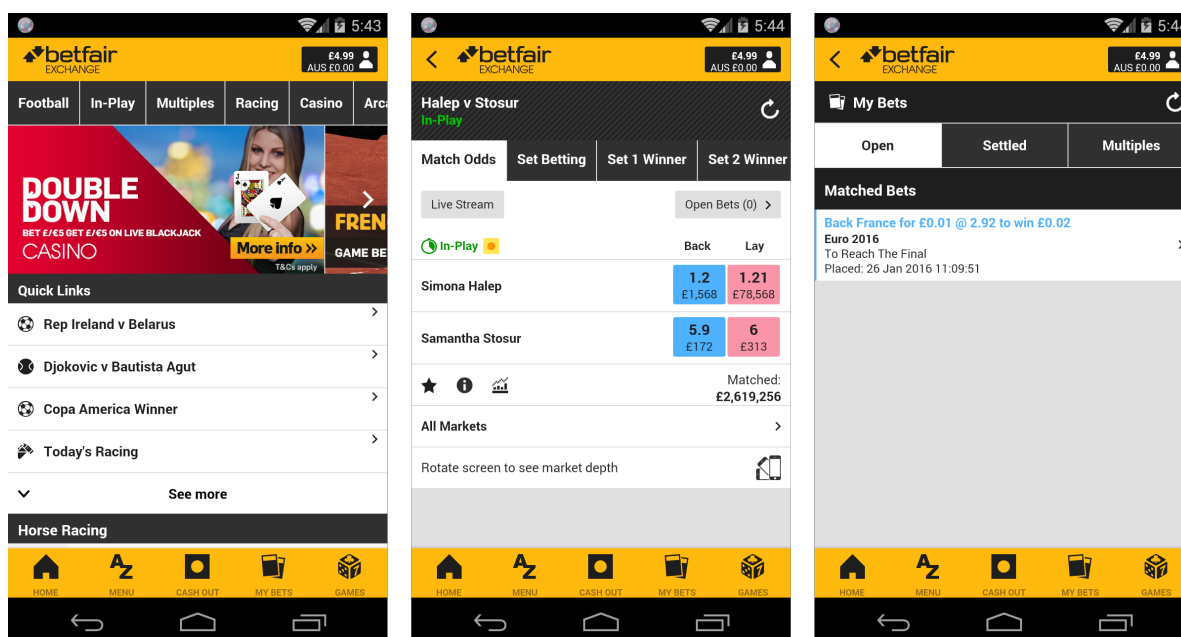


Figure 3.2: Betfair application screenshots.

3.3 Conclusions

The comparison criteria applied in the analysis of the different applications available in the market are as follows:

- **Place Bets** - place bets in different markets;
- **Price Infos** - Display all the price information in the different available markets;
- **Grid Interface** - Show the market information in Grid View;
- **Ladder Interface** - Show the market information in Ladder View;
- **Multibet Interface** - Show the market information in Multibet View;
- **Dutching**[22] - Apply dutching techniques for the selections in a market;
- **Bookmaking**[23] - Apply bookmaking techniques for the selections in a market;
- **Inplay markets** - Access all markets inplay in Betfair;
- **Live Scores** - Live Scores information for events;
- **Live Stream** - Display a stream to watch the event live;
- **Account Information** - Display information regarding the Betfair account (Currency, Balance,..);
- **Multiple Wallets** - Manage different wallets;
- **Free** - if the application is free to use.

In the below table (3.1) it is presented a comparison between the principal features of all applications. To represent the verification of a characteristic it is used the symbol ✓ or ✗, otherwise.

	BetOnDroid	Betfair Mobile	Traderline Mobile
Place bets	✓	✓	✓
Price infos	✓	✓	✓
Grid Interface	✗	✓	✓
Ladder Interface	✓	✗	✓
Multibet Interface	✗	✗	✓
Dutching	✗	✗	✓
Bookmaking	✗	✗	✓
In-play markets	✓	✓	✓
Live Scores	✓	✓	✓
Live Streaming	✗	✓	✗
Account Information	✓	✓	✓
Multiple Wallets	✓	✓	✓
Free Software License	✓ ³	✓	✗ ⁴

Table 3.1: Comparison table between different Betfair registered Betting Exchange applications.

Mobile betting applications[24] are the future, and the bettors are looking for applications that are easy to use, with attractive interfaces, reliable and other tools that help in a more effective Trading experience.

Being in the field and able to make bets makes a great advantage when compared to other bettors that are watching, for example, a game in the TV, where there is always a slight delay. Online betting is all about opportunity and timing, and delays can make all the difference between finishing a Trading session with a positive outcome, or the other way around.

The principal difference between Traderline and the remaining applications is the possibility to apply automatically advanced strategies such as Dutching and Bookmaking, but in order to apply these strategies it is required a great amount of calculations, and before those calculations are made, the odds are likely to change, making it necessary

³Free version contains restrictions:Ads. No In-play markets. No livescores.

⁴Requires a license to use.

do it all over again, endlessly, which is not practical at all. This is why having mobile software that automatises this process is so important and useful.

Live Streaming is a feature that could be added to Traderline, however it could break the Trading Experience, since mobile phones are small size devices, in order to live stream a video and keep all the information necessary for Trading is quite difficult in the same page, in Traderline a leverage is given for market information.

Traderline's interface was designed to enrich trading experience, with all the different Views (Grid, Ladder, Multibet), allowing different bettors to choose their favourite way to follow up the markets.

4

Technical Approach

This section aims to give an insight about the technology used to build the Traderline application. Xamarin Forms is the tool used to develop the mobile application for Traderline, it was imposed as a constraint, hence there was no need to evaluate other tools available or generate a comparison table between possible candidates.

Despite being a technological constraint, it is important to understand why this technology was chosen for the Traderline Solution. As previously explained, Traderline is a multi-platform solution for sports trading, so it has to run across different devices with completely different operating systems: Windows, Mac, iOS, Android. To develop such applications natively, implied that different projects had to be developed and maintained. Different projects that would be built by different people, with different approaches to problems which would imply different implementations. The cost of maintaining these applications would also be a pain, the experience from the previous Traderline applications led to a new process and new technology: a technology that would allow to build once, run everywhere, to share the biggest codebase possible and ease the development and maintenance of the different platforms. Xamarin is nowadays the biggest player in this market and the perfect solution for Traderline.

4.1 Xamarin

Xamarin is a cross-platform mobile application development solution that creates a unified environment for developers[25]. Xamarin Framework distinguishes itself from other cross platform tools mainly because of its hardware features, the concept of code sharing and native bindings. These factors contribute to good performance results and native user experience, which constitutes as a great difference to many other cross-platform tools. To develop mobile apps with Xamarin there are mainly two different approaches: Xamarin.iOS / Xamarin.Android, or Xamarin.Forms. Later in this section will be explained the main differences and which was the selected option for this project.

Xamarin is a port of .NET to the iOS and Android operating systems with support for Windows Phone. Xamarin.iOS is based on MonoTouch and Xamarin.Android on Mono. Xamarin offers C# bindings for the native Android and iOS APIs, allowing developers to take advantage of the Android and iOS user interface, graphics, notifications, animations, and phone features such as location and camera—all using C#. Everytime there is an update of the Android and iOS APIs, Xamarin releases an update to match all the new features.

Xamarin.Forms allows developers to build native interfaces for both Android and iOS from a single project, developed in C#.



Figure 4.1: Xamarin C# libraries bindings to native OS SDKs and .NET [1]

4.1.1 Hardware Features

Xamarin provides tools native bindings to iOS and Android SDK, allowing a direct access to all native methods, but in a different language (C#). This is extremely important since developers that are common with the native languages can easily develop mobile applications with Xamarin.

The supported features are:

- Contacts
- Geolocation
- Compass, Gyroscope and Accelerometer
- Video and Audio
- Notifications

4.1.2 Code Sharing

The main purpose of building cross-platform applications is being able to share code across the applications running on different platforms. In Xamarin, there are two possible approaches:

- Shared Projects - Share a project between the different platforms, using compiler directives for platform-specific code.
- Portable Class Libraries - this approach is different from the Shared Project option. There is no direct access for platform-specific code, only through interfaces.

4.1.3 Native Bindings

Native bindings consist of accessing the native APIs, from iOS or Android, from a Xamarin mobile application. All the bindings are ported to C#.

4.1.4 Xamarin Forms

Xamarin.Forms is a toolkit of cross-platform UI classes providing an abstraction from the other possibilities to develop for mobile devices: Xamarin.Android and Xamarin.iOS. Xamarin.Android and Xamarin.iOS provide mapped classes to their respective native UI SDKs: iOS UIKit and Android SDK. Xamarin.Forms also binds directly to the native Windows Phone SDK. This provides a cross-platform set of UI components that render in each of the three native operative systems[1].

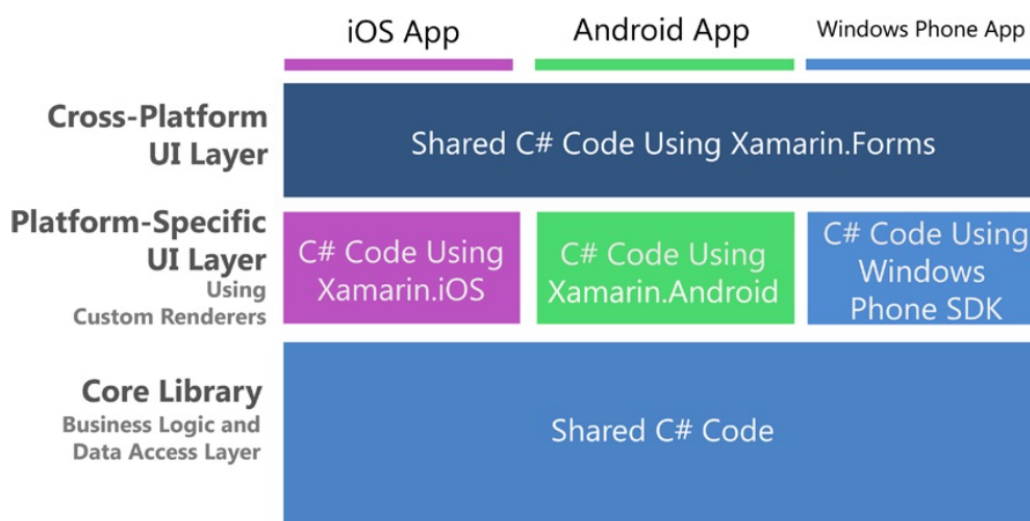


Figure 4.2: Xamarin Forms architecture [1]

Xamarin.Forms is a great tool to begin building an app quickly. It provides a cross-platform toolkit of pages, layouts and controls that are built with Extensible Application Markup Language (XAML) or coded in C#, using Page, Layout, and View classes. This API is extremely helpful, it provides a brand range of built-in cross-platform mobile UI patterns, such as Pages, Layouts, Controls, Views. The following image (4.3) illustrates the different Pages available in Xamarin.Forms.

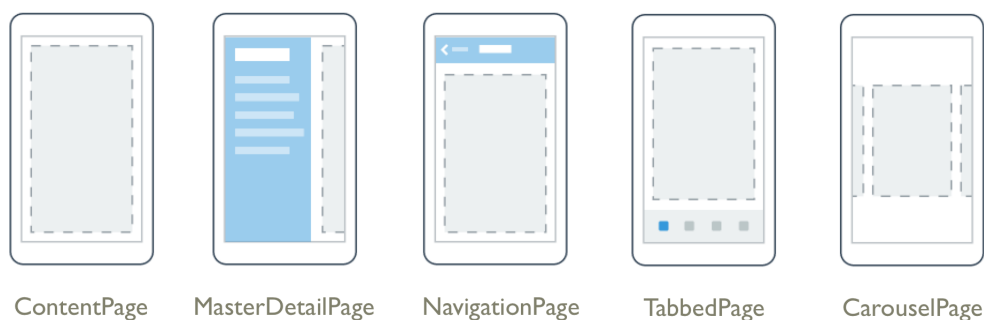


Figure 4.3: Xamarin.Forms Pages represent cross-platform mobile app screens [2].

Layouts represent the standard formats used on the different mobile platforms. Used within those layouts are the controls, or views, like buttons or lists. Many of these views have built-in data binding options. The following image (4.4) represents the variety of Layouts that can be used to compose user interface controls into logical structures.

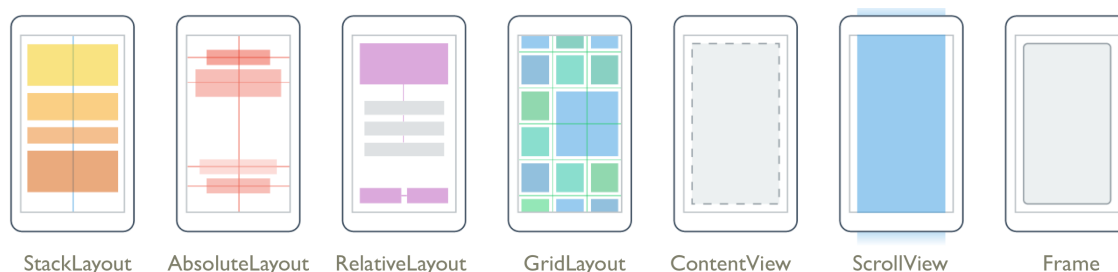


Figure 4.4: Xamarin.Forms Layouts [2]

Xamarin.Forms comprises platform-independent classes that are bound to their native platform-specific counterparts. This means it is possible to develop basic, native UIs for all three platforms with almost no knowledge of iOS and Android UIs.

Although this is a great advantage compared to many other cross-platform tools, with great power comes great responsibilities. The process of debugging is extremely painful, since there are so many layers of abstraction. There is also the problem of extensibility, being restricted to the base elements of Xamarin.Forms, if we intend to use something different, we need to make Custom Renderers, or sometimes Specific Platform code.

4.2 Conclusions

Mobile applications have been increasing in the latest times, the easy access to information from a mobile device is an important part of people's lives.

It has therefore become a challenge to create mobile applications that can run on many common mobile operating systems like iOS, Android. One approach to this problem is to have separate code bases and separate development teams for each platform for the same application, which is obviously a costly solution. In order to reduce development effort and cost while providing rapid delivery to all application markets at once, a preferred alternative is to use a cross-platform mobile application development tool/framework[26]. Xamarin is a great tool for this problem, and with the recent Xamarin Forms release, it allows the biggest shared code base possible. If we think about maintenance, if a problem is discovered, the solution will be reflected in the different platforms. The cost of maintenance and development is reduced compared to the usual software development for different platforms.

In the previous Traderline versions, the development of each platform (Desktop/Mobile) was separate. This meant that, whenever a bug was detected in a platform, if it was also present in the other one, it had to be corrected twice, the maintenance of the applications was costly. Using Xamarin allows a Core Module to be shared across all platforms, if the same situation occurs, when fixing it for one platform, it automatically is fixed for all platforms, with less costs.

5

Requirements Analysis

Requirements definition is of major importance to the success of a software project, because they give a good notion of what needs to be done and how long it might take to do it, they are the basis during the entire development stage.

The main features of the Traderline solution are as follows:

- Login;
- Select and search Markets;
- Real-time visualisation of markets of different sports;
- Bets in Grid mode;
- Bets in Ladder mode;
- Bets in Multibet mode;
- Management of favourite sports;
- Consult account information;
- Application Configurations;
- License registration;
- User Manual;
- Phone and Tablet support.

The process of gathering requirements involved the analysis of the previous Traderline Application, and then extracting the list of features, as well as adding additional features. The previous Traderline Mobile application had its flaws, several bugs that were not corrected, the lack of mobile “touch and feel” actions, so it was important, in the first place, to certify that all the features present in the previous version, were correctly implemented in the new version. After this, improvements to the design and user experience, and, above all, an application with quality.

An existing System Interface Analysis was conducted, where an analysis of the user interface is the black-box equivalent of reverse engineering the source code, by mapping out the behaviours that the application exhibited in response to inputs events. By using this process, flow-diagrams were designed that facilitated the comprehension of all

features. After this analysis, a meeting with the stakeholders was set up to discuss and clarify all the requirements and eventually new requirements were discovered.

In order to prioritise work, MoSCoW[27] scale was used. This scale divides the requirements into four categories: *Must*, *Should*, *Could* and *Won't*. This prioritisation allows a better understanding of what are the features to be implemented first (*MUST*), these consisted of all the requirements of the previous application, whereas requirements categorised as (*COULD*) are new requirements that may or may not be implemented.

In this chapter it is presented a summary of the requirements specification and in appendix the Flow Diagrams.

5.1 Functional Requirements

In this section will be presented the functional requirements for the Traderline Mobile application. These functional requirements are presented from a broad point of view, as it is not within the scope of this document to do a full requirements specification.

The Functional Requirements were divided by the pages or views of the application. This division allowed a separation of requirements and a better understanding of the application.

5.1.1 Requirements for Login

<i>User Story</i>	<i>Summary</i>	<i>Priority</i>
LOGIN-01	Enable/Disable Real/Practice Mode	MUST
LOGIN-02	Enable/Disable Remember me option	MUST
LOGIN-03	Enable/Disable Australian Markets	MUST
LOGIN-04	Register account in Betfair	MUST
LOGIN-05	Retrieve password	MUST
LOGIN-06	Login with valid Betfair credentials	MUST

Table 5.1: Requirements specification for Login Page

5.1.2 Requirements for Favourite Sports

<i>User Story</i>	<i>Summary</i>	<i>Priority</i>
FAVSPORT-01	Show list of favourite sports	MUST
FAVSPORT-02	Navigate to list of events for the chosen sport	MUST

Table 5.2: Requirements specification for Favourite Sports Page

5.1.3 Requirements for General Options

<i>User Story</i>	<i>Summary</i>	<i>Priority</i>
GENOPT-01	Enable/Disable Inclusion of Australian Markets	MUST
GENOPT-02	Change API Request Modes: Saving, Normal, Intensive	MUST
GENOPT-03	Change pre-defined market view: Grid, Ladder or Multibet	MUST
GENOPT-04	Enable/Disable cancel unmatched bets	MUST
GENOPT-05	Enable/Disable Show hedge value in ladder	MUST
GENOPT-06	Change Ladder column order, Lay-Back or Back-Lay	MUST
GENOPT-07	Enable/Disable inverted values in ladder	MUST
GENOPT-08	Enable/Disable show fractional odds	MUST
GENOPT-09	Enable/Disable show live results	MUST
GENOPT-10	Enable/Disable show live results	MUST
GENOPT-11	Enable/Disable hide inferior menu in market page	MUST
GENOPT-12	Enable/Disable hide account balance	MUST
GENOPT-13	Enable/Disable application sounds	MUST
GENOPT-14	Manage favourite sports	MUST

Table 5.3: Requirements specification for General Options Page

5.1.4 Requirements for All Sports Page

<i>User Story</i>	<i>Summary</i>	<i>Priority</i>
ALLSPORT-01	Show list of all sports available	MUST
ALLSPORT-02	Navigate to the leagues of the chosen sports	MUST

Table 5.4: Requirements Specification for All Sports Page

5.1.5 Requirements for Search Page

<i>User Story</i>	<i>Summary</i>	<i>Priority</i>
SEARCH-01	Search event by name	MUST
SEARCH-02	Display event information	MUST
SEARCH-03	Navigate to market	MUST

Table 5.5: Requirements Specification for Search Page

5.1.6 Requirements for Main Page

<i>User Story</i>	<i>Summary</i>	<i>Priority</i>
ACCOUNT-01	Show account balance	MUST
ACCOUNT-02	Show current exposure	MUST
ACCOUNT-03	Show API Lag	MUST

Table 5.6: Requirements Specification for Account View

<i>User Story</i>	<i>Summary</i>	<i>Priority</i>
LIVE-01	Show list of live available events by sport	MUST
LIVE-02	Show livescore, if available	MUST
LIVE-03	Navigate to event market page	MUST

Table 5.7: Requirements Specification for Live View

5.1.7 Requirements for Market Interface

<i>User Story</i>	<i>Summary</i>	<i>Priority</i>
MARKET-01	Show overall event score	MUST
MARKET-02	Show event name	MUST
MARKET-03	Show event status	MUST
MARKET-04	Show my account	MUST
MARKET-05	Show menu to navigate to Bets, Stats, Grid, Ladder, Multibet	MUST
MARKET-06	Trade confirmation	COULD

Table 5.8: Requirements Specification for Market

<i>User Story</i>	<i>Summary</i>	<i>Priority</i>
STAK-01	Choose from 3 default stakes	MUST
STAK-02	Input stake value	MUST
STAK-03	Set bet staking plan (Stake, Liability, Profit)	MUST

Table 5.9: Requirements Specification for Stakes

5.1.8 Requirements for Grid View

<i>User Story</i>	<i>Summary</i>	<i>Priority</i>
GRID-01	Show best back and lay odds	MUST
GRID-02	Show profit and loss per selection	MUST
GRID-03	Trade current bet	MUST
GRID-04	Show cashout value position	MUST
GRID-05	Swipe to open list of markets	COULD

Table 5.10: Requirements Specification for Grid View

Requirements for My Bets

<i>User Story</i>	<i>Summary</i>	<i>Priority</i>
BETS-01	List matched bets	MUST
BETS-02	List unmatched bets	MUST
BETS-03	Cancel one unmatched bet	MUST
BETS-04	Cancel all unmatched bets	MUST
BETS-05	Edit bet odds	MUST
BETS-06	Edit bet stake	MUST

Table 5.11: Requirements Specification for My Bets View

Requirements for Statistics

<i>User Story</i>	<i>Summary</i>	<i>Priority</i>
STATS-01	Show Betfair chart for selected market	MUST
STATS-02	Change market	MUST
STATS-03	Invert chart axis	MUST

Table 5.12: Requirements Specification for Statistics Page

Requirements for Ladder View

<i>User Story</i>	<i>Summary</i>	<i>Priority</i>
LADDER-01	Change market	MUST
LADDER-02	Fullscreen mode	MUST
LADDER-03	Center odds	MUST
LADDER-04	Trade bets	MUST
LADDER-05	Swipe to open list of markets	COULD
LADDER-06	Multi ladder	COULD

Table 5.13: Requirements Specification for Ladder View

Requirements for Multibet View

<i>User Story</i>	<i>Summary</i>	<i>Priority</i>
MULTI-01	Dutching mode	MUST
MULTI-02	Bookmaking mode	MUST
MULTI-03	Show total stake	MUST
MULTI-04	Swipe to open list of markets	COULD

Table 5.14: Requirements Specification for Multibet View

5.1.9 Requirements for Bet Settings Page

<i>User Story</i>	<i>Summary</i>	<i>Priority</i>
BETSET-01	Edit Tick offset	MUST
BETSET-02	Enable/Disable stop loss	MUST
BETSET-03	Enable/Disable bet without confirmation	MUST
BETSET-04	Change bet persistence mode	MUST

Table 5.15: Requirements Specification for Bet Settings Page

5.1.10 Requirements for Registration page

<i>User Story</i>	<i>Summary</i>	<i>Priority</i>
REG-01	Name, e-mail, country	MUST

Table 5.16: Requirements Specification for Registration Page

5.1.11 Requirements for Update System

<i>User Story</i>	<i>Summary</i>	<i>Priority</i>
UPDATE-01	Application update without uninstall and losing user preferences	COULD
UPDATE-02	Application can only be executed with the latest version	COULD

Table 5.17: Requirements Specification for Update System

5.2 Non-Functional Requirements

Non-functional Requirements, represent how a system should operate, rather than specific behaviours. In order to understand the Non Functional Requirements necessary for the Traderline application, a meeting with the stakeholders was conducted in order to understand how the application should present itself, as this poses as an important step to define the architecture of the overall system. There were several aspects pointed as important to ensure the product quality, a list was defined and later refined, gathering a total of four non-functional requirements, To clarify these attributes a list of scenarios are presented.

5.2.1 Security

Preserving user information is of an extreme importance, since this is a Betting application, with real money involved, it is critical to ensure that all the communications are secure against possible vulnerabilities.

<i>ID</i>	<i>Type</i>	<i>Summary</i>	<i>Priority</i>
RNF-01	Security	The system shall encrypt the user information, as well as all the communications must be secure	MUST

Table 5.18: Non-Functional Requirement 1 - Security

5.2.2 Extensibility

Extensibility is an approach whereby the architecture, design and implementation of a system actively consider and cater for future change[28]. At any time new functionalities to be added to Traderline may arise, as well as modifications may be requested to improve Traderline Mobile experience. Taking this into consideration, the system should be modular and well specified, so that adding new features will not take a great deal of impact in the development process.

<i>ID</i>	<i>Type</i>	<i>Summary</i>	<i>Priority</i>
RNF-02	Extensibility	The system shall be extensible to allow the inclusion of new features in no longer than 5 days	COULD

Table 5.19: Non-Functional Requirement 2 - Extensibility

5.2.3 Usability

Usability refers to the ease with which a user can learn to operate, prepare inputs for, and interpret outputs of system or component[29]. As a mobile application, and specially, a betting mobile application as it is Traderline, it is important that the system reacts quickly to the user input, the look and feel of the application is attractive to its users.

<i>ID</i>	<i>Type</i>	<i>Summary</i>	<i>Priority</i>
RNF-03	Usability	The user shall be able to perform an action with a number of errors that will not exceed 8	MUST

Table 5.20: Non-Functional Requirement 3 - Usability

5.2.4 Performance

Mobile application performance is defined by the user's perception of how well the application performs. This means that the performance of your application is measured by how responsive it is, how quickly it starts up, how well it uses device memory, how well it uses device power[30]. Despite dealing with a great amount of information which is constantly being updated and several business transactions, Traderline Mobile must ensure its users that they can perform their tasks quickly and efficiently.

<i>ID</i>	<i>Type</i>	<i>Summary</i>	<i>Priority</i>
RNF-04	Performance	The user shall be able to open a Market within 6 seconds	MUST

Table 5.21: Non-Functional Requirement 4 - Performance

6

Solution

This chapter aims to clarify the Traderline solution. Will be presented the used technologies, an overview of the system architecture, the project structure and developed work, some analysis between the Traderline Mobile application versions and the conducted tests for the application validation.

6.1 Technologies

In order to develop the Traderline mobile application, several tools and technologies were used. Most of these technologies are related to Xamarin, since it was, like previously said, a constraint imposed by the client. Here is presented a list of the relevant technologies and its importance for the development:

- **Xamarin Studio:** sophisticated IDE with many features for creating Xamarin.iOS, Xamarin.Mac, Xamarin.Android, and Xamarin.Forms applications [31]. It includes a rich editor, debugging, native platform integration with iOS, Mac and Android, and integrated source control. An essential tool for the development of Traderline Mobile application.
- **Git:** widely-used version control system for software development. It is a distributed revision control system with an emphasis on speed, data integrity, and support for distributed, non-linear workflows[32]. The flow adopted in the Traderline development team involved two principal branches, the master, where all the tested features would be merged, and the development, for all the new, and untested features.
- **C#:** multi-paradigm programming language encompassing strong typing, imperative, declarative, functional, generic, object-oriented (class-based), and component-oriented programming disciplines[33]. It is the base language used in this project.
- **XAML:** declarative XML-based language that allows the definition of user interfaces in Xamarin.Forms applications using markup rather than code[34]. It was not necessary to program all the user interfaces using XAML, however it is more succinct and visually coherent, that's why it was adopted in this project. It is also recommended to use with the MVVM architecture, which was adopted for the Traderline Mobile solution: XAML defines the View that is linked to ViewModel code through XAML-based data bindings.

- **SQLite[35]**: embedded database software for local/client storage in application software. It is used to store the user preferences in the application.

6.2 Architecture

In this section, the system's architecture is presented according to different levels:

- **Level 0** - understand the overall system, the interactions between the user, the application and external systems
- **Level 1** - approaches the high-level shape of the software architecture and how responsibilities are distributed across it.

This separation of concerns focuses on a visual communication of the software architecture [36].

6.2.1 Level 0

The Level 0 represents all external entities that may interact with the system. In this case, we have a User, who uses the Traderline Application. The Traderline application interacts with external services, the Betfair API and Stratio Soccer API.

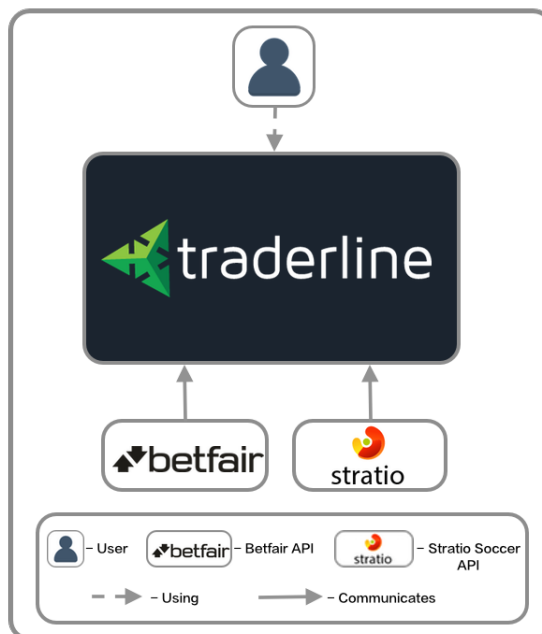


Figure 6.1: Level 0 architecture diagram

The Betfair API provides all the betting information, such as markets, odds, events and the Soccer API provides information about the Soccer events, like live scores and statistics. In order to use the Betfair API it is necessary an application key, the application must be registered to the Betfair API Program, and a session token, which is given after the login with valid betfair credentials. All the following requests are validated with this session token.

The Stratio Soccer API feeds the Traderline Application with live soccer results.

6.2.2 Level 1

After understanding the system's interaction with external entities, it is now presented the modules that compose the Traderline Application in the following image 6.2 .

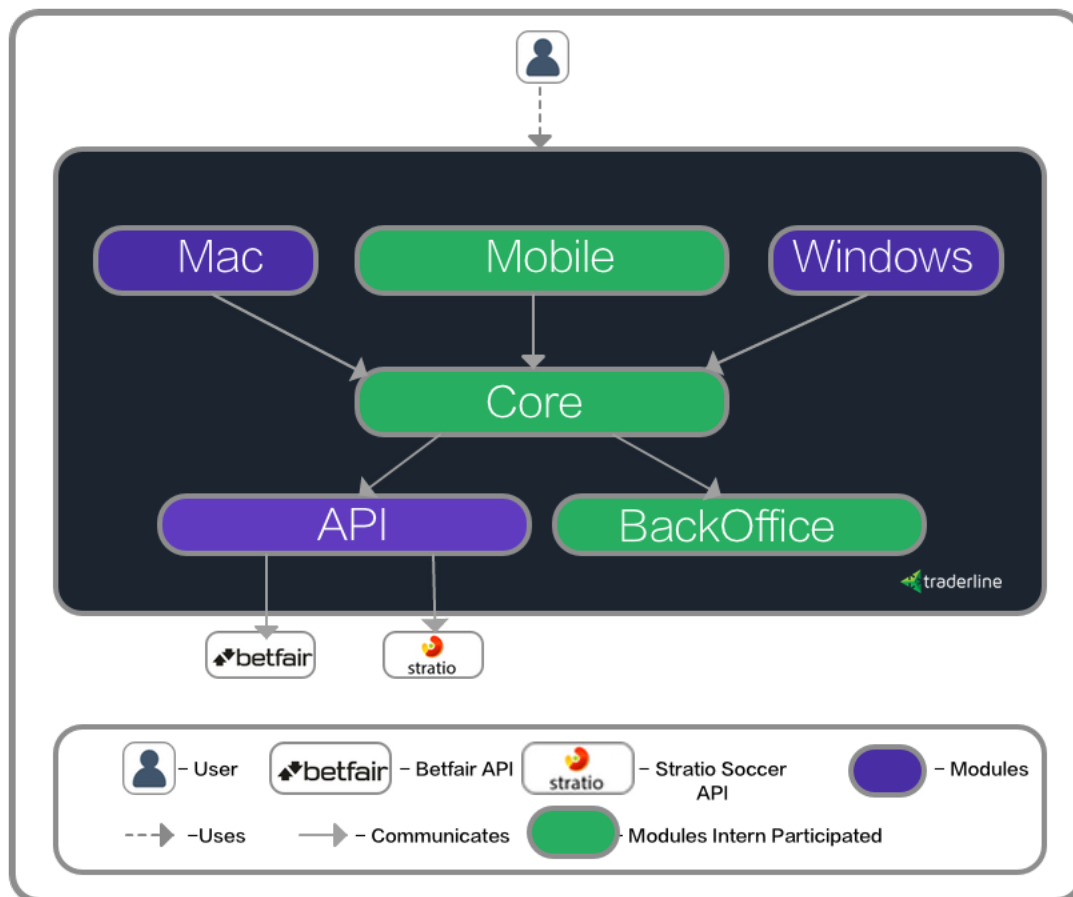


Figure 6.2: Level 1 architecture diagram

The Traderline Solution is separated into six different modules: API, BackOffice, Core, Mobile, Windows and Mac. In green are represented the Modules whose development the intern participated. The principal responsibility is the Mobile Module, regarding the Traderline Android application, however, since this is a cross-platform solution, the intern actively participated and interacted in the architecture and definition of other Modules.

The API Module is responsible to interact with the API's of Betfair and Soccer. This Module acts as a intermediary between Core and external API's. It serves the Core with all the information necessary for the system. Despite not being part of the development of this Module, the intern several times used this for debugging purposes, to check the information from each request to detect the causes of some problems.

The BackOffice Module has the information regarding Traderline licenses, and it was also developed a service for the custom update system for the Android Application. This service is specific for the Mobile Module, it gives information, through a http request, regarding the latest version of the application as well as a download link for the newest version.

The Core Module stores all the information that is received from the API, it is also responsible for the business logic and data models. This is the main module of Traderline,

and, once again, the intern actively participated in this definition. During the development of the Traderline Mobile, there were several Models, or services regarding User Interface, such as notification system, enumerations that were suggested to be part of the Core Module, rather than being present only in Mobile. This decision was made thinking of one of the objectives of Traderline, code-sharing, and since this features would be most likely also implemented in the Mac and Windows application, it would make sense to have common interfaces for all the different visualisation modules. The business logic is mainly implemented in this module, but since the Traderline Mobile development was faster than the Desktop Module, the major bugs were discovered in the Mobile application. The intern always participated in the debugging process with the Core Development team to solve each problem.

The Mobile Module has the Android and iOS applications. The purpose of this internship is the Traderline Mobile Android application, and it was the focus during the development, but, since the Technology used was Xamarin and most specifically Xamarin.Forms, with little effort it is possible to build the application for iOS platform.

The Mobile module consists of the presentation of data, user interface. It has also Models to use the information from the Core Module, as well as implementation of some business logic. There is also the Database Model, that consists of each user preference and where the user information is properly encrypted to match Betfair's requirements.

6.3 Project Structure

In this section, the current Project Structure for developing the Traderline mobile application will be explained. As it has been previously referred, the main purpose of developing with Xamarin technology is the Code Sharing Strategy. As we can see in the figure below (6.3), we have the Modules explained in the Architecture Section, the API Modules, the Core Module and the Presentation Module, including the Mobile. The API Module is referenced in Core Module, whilst Core is referenced in the Mobile. This allows a separation of concepts, the business logic and the services are present in the Core Project, the Mobile is only responsible to access them.

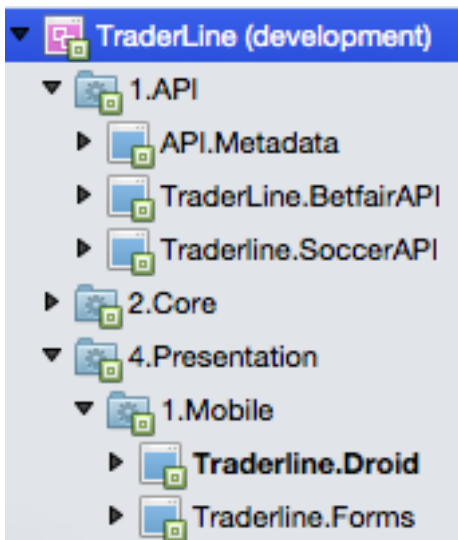


Figure 6.3: Traderline Solution Structure

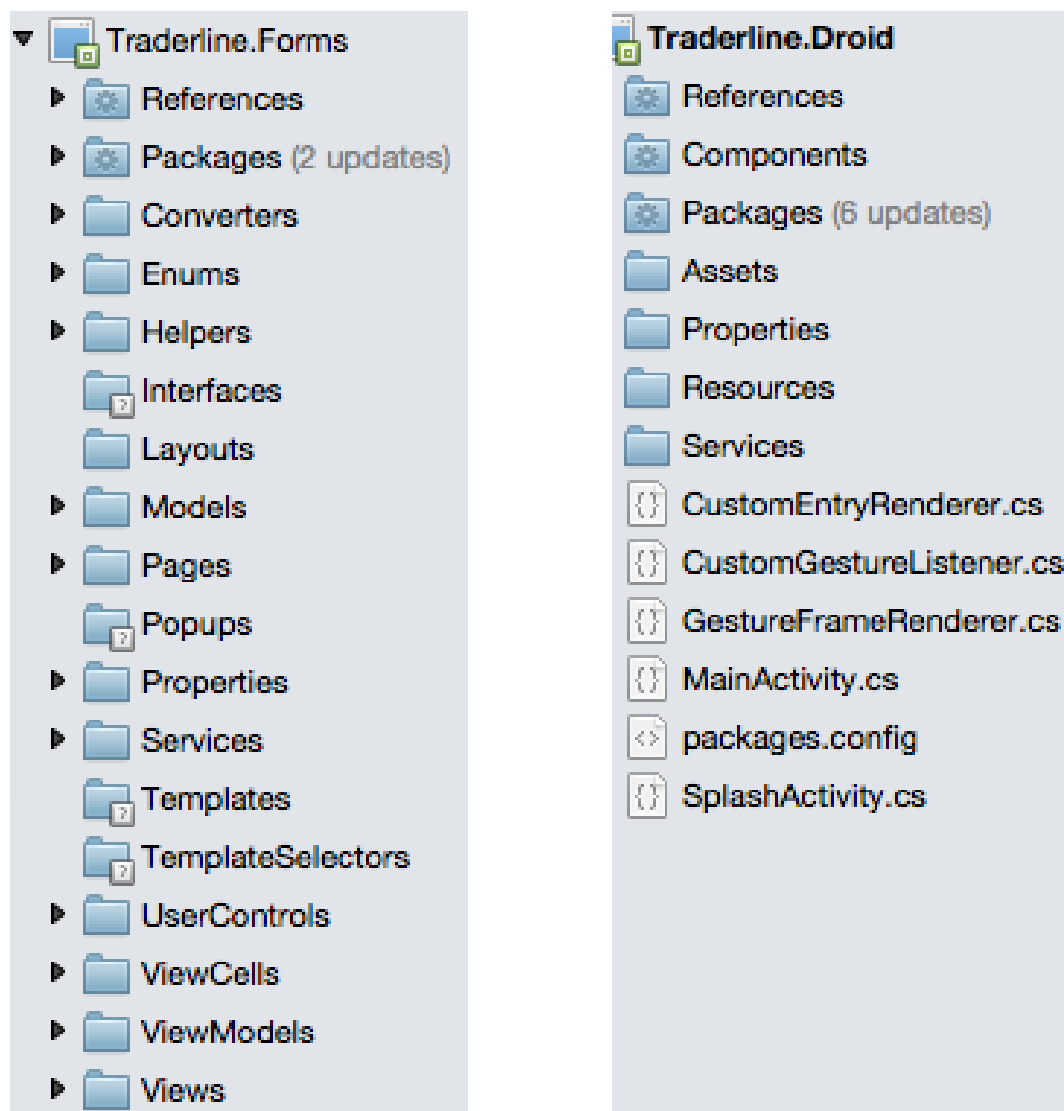


Figure 6.4: Traderline.Forms Project Structure Figure 6.5: Traderline.Droid Project Structure

If we zoom in the Mobile Project, it is composed of two parts: Traderline.Droid and Traderline.Forms. In the Traderline.Forms we develop all the code regarding the Android application, only when it is necessary to have a custom renderer it goes to the Traderline.Droid (6.5).

The purpose of using Forms is, once again, the code sharing, this is why few code is platform-specific. The structure within Traderline.Forms can be seen above (6.4). The Xamarin Forms provides a suit of components to help develop a mobile application, Pages, Views, ViewCells. Every application screen consists of a Page. In this Page we have ContentView to display Content. Within each ContentView we might have different layouts: StackLayouts, Absolute Layouts, that help position the elements, such as lists, tabs, grids and controls like buttons.

6.4 Results

In this section will be presented the accomplished results of this internship. These results correspond to the implementation of all the requirements planned for the Traderline Mobile application, even the requirements categorised as *COULD* or *SHOULD*, fulfilling all the objectives. After presenting the Traderline Mobile application, will be also presented an analysis between the old traderline mobile application and the one developed in this internship.



Figure 6.6: Traderline Mobile application (Phone and Tablet)

6.4.1 Implemented Requirements

This section aims to present the Traderline Mobile application, displaying several screenshots of the application and the main features.

After installing the Traderline app, and once it is opened, the first screen is the Login (Figure 6.7). Here the user can insert valid Betfair credentials and enter the application either in Real Mode (dealing with actual money), or practice mode (no money involved). There is also the possibility to remember username, enable Australian Markets, retrieve password, or create a new Betfair account. After validating the credentials, the application navigates to the Home Screen. From this area it is possible to access all the different menus and features offered by the application.

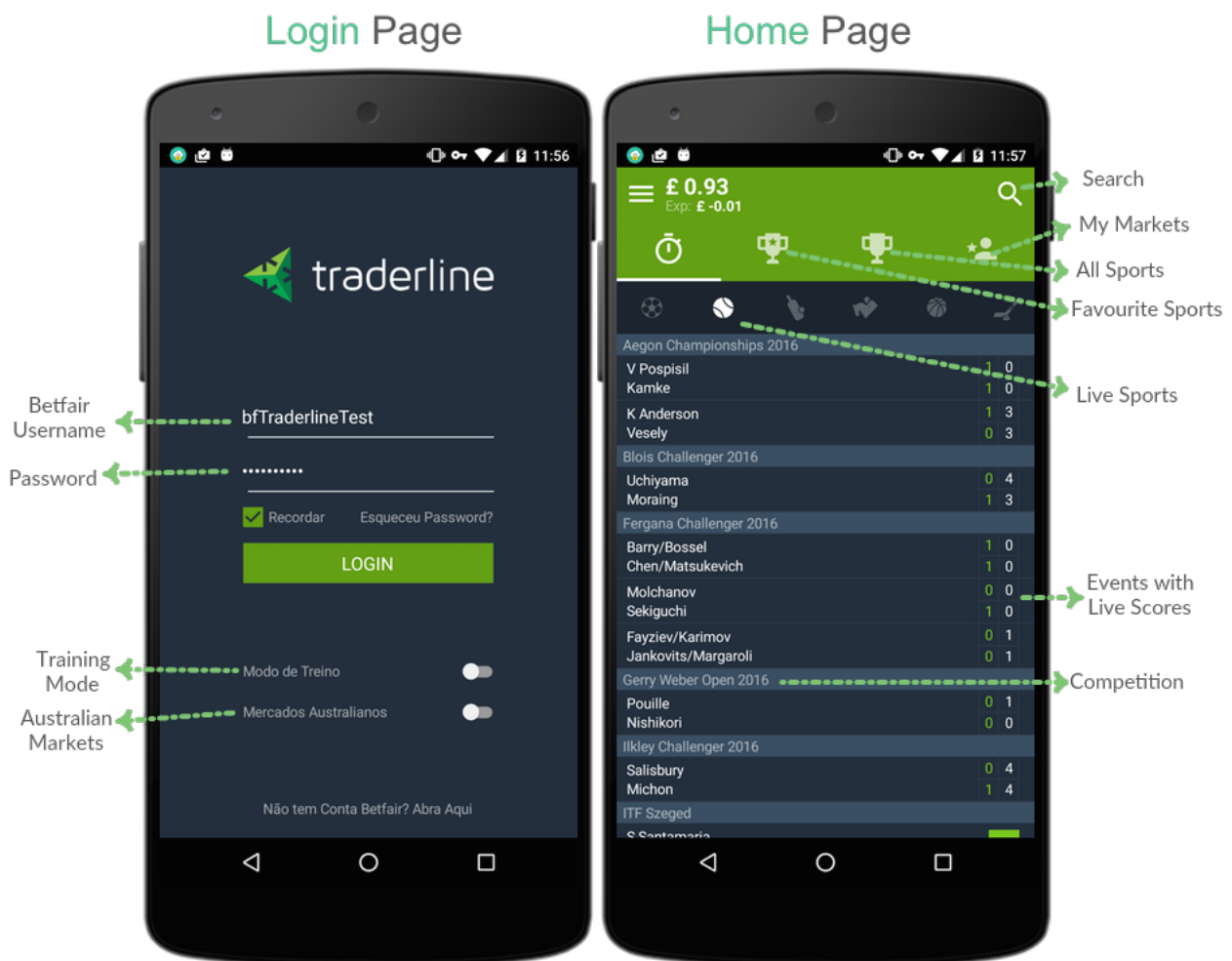


Figure 6.7: Login and Home Page Screenshots

Home is split in four main areas: Live Events, Main Sports, All Sports and My Markets. In Live Events view, all the sport events that are live, are displayed, organised by sport. If the livenesscores are available for the event, it is also shown in the corresponding event cell. In the All Sports View, it is possible to choose the sport where the user intends to bet through a market tree interface. By touching the sport, we can access the main leagues, divisions and tournaments and we can see all the available markets for placing bets. This tree interface gives easy access to any match and any sport, even if the matches won't be played in the same day. It is possible to access the search view from almost

every screen in the Traderline Mobile applications. This allows the user to quickly find and navigate to a specific market. The All Sports and Search View are illustrated in Figure 6.8

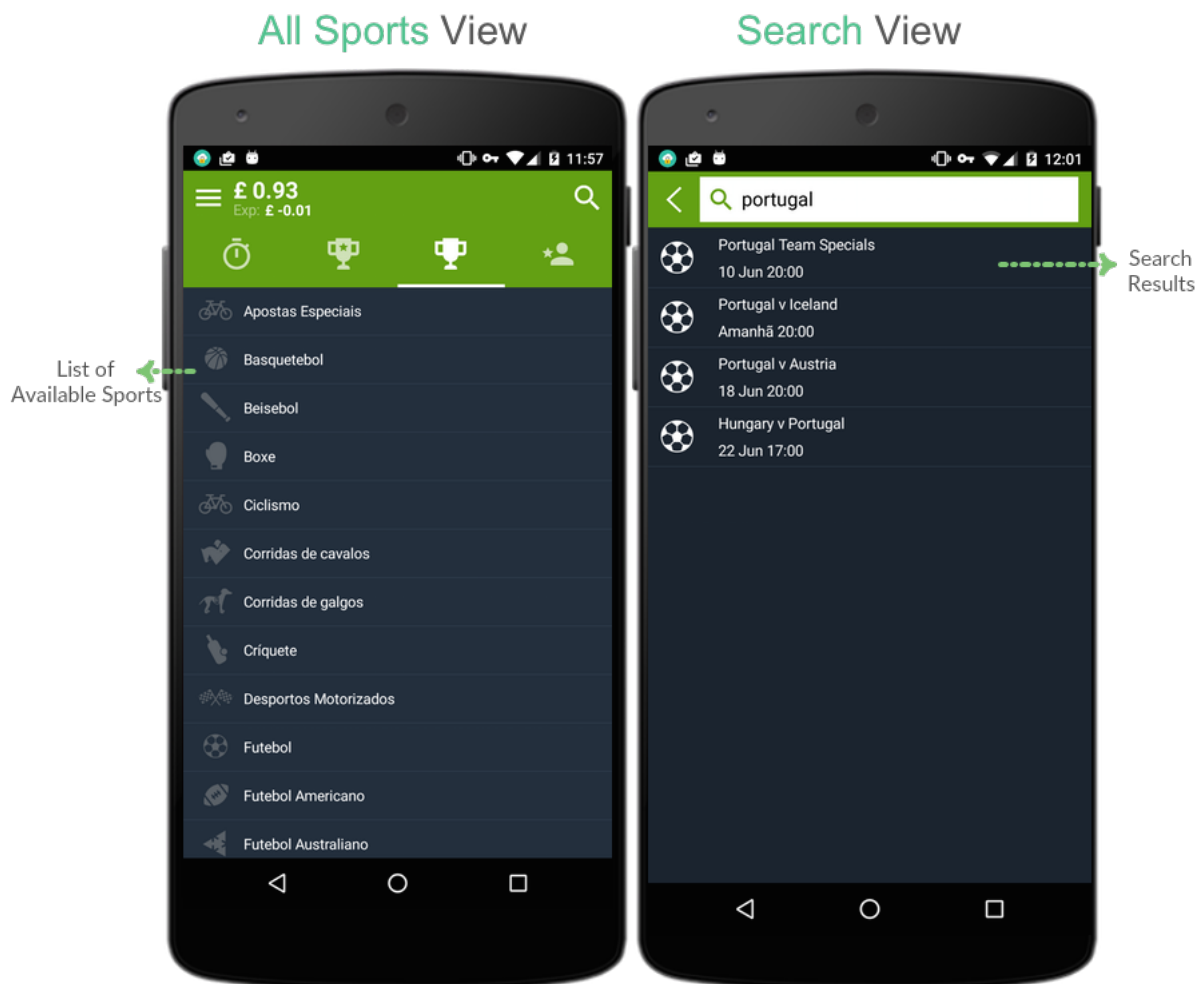


Figure 6.8: All Sports and Search View Screenshots

When entering a market, Traderline is pre-set to display the grid interface. This is similar to Betfair's standard interface. The odds shown in the grid are the actual odds in the market, with the respective liquidity values below each odd. Another way to analyse a market is through the Ladder Interface. This interface, besides changing the way the odds of the market are displayed, also provides additional information on the market, which is useful for a more in depth analysis of each odd's liquidity and their predictable trends of evolution. Figure (6.9) represents the Grid and Ladder Views inside a Market.



Figure 6.9: Market Views (Grid, Ladder) Screenshots

The access to Multibet Interface is through the top right menu of the market. In this interface, we can place several bets simultaneously in the selections of the market, through the Dutching or Bookmaking modes. These betting modes drastically reduce the risk and uncertainty associated to the bets, because the many possible outcomes of a match are covered, so that you can always make a profit.

- **Dutching Mode** - place several back bets
- **Bookmaking Mode** - place several lay bets

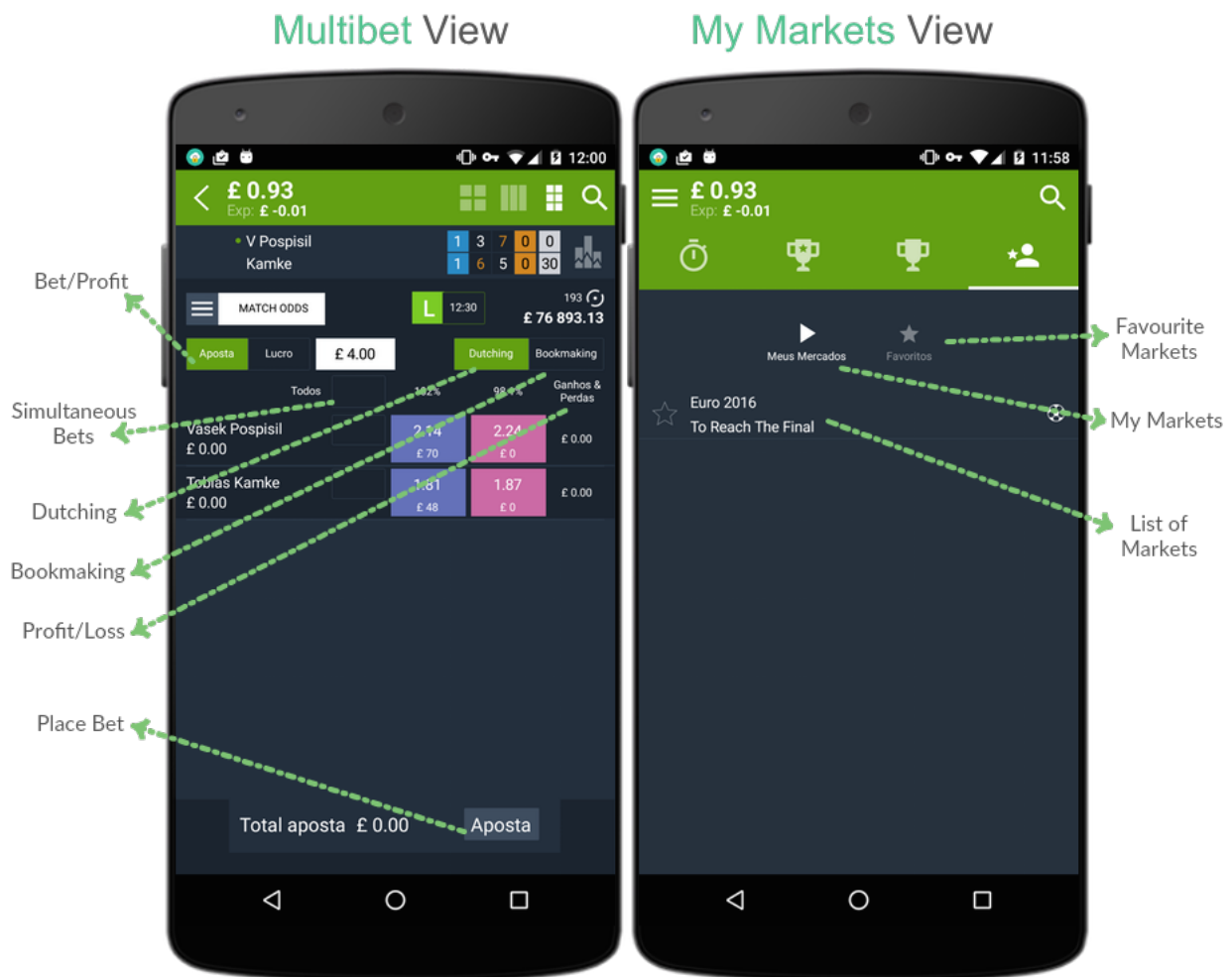


Figure 6.10: Multibet and My Markets View Screenshots

Figure 6.10 represents Multibet and My Markets View. The My Markets menu is split in two windows: My Markets and Favourites.

- **My Markets** - list of market with current bets
- **Favourite Markets** - list of markets that were manually added to the list of favourites

By accessing the Main Sports View in Home, it is possible to select one of the favourites sports and filter all the upcoming events for that sport. In the case of football it is also possible to quickly filter events by the Principal Football Leagues.

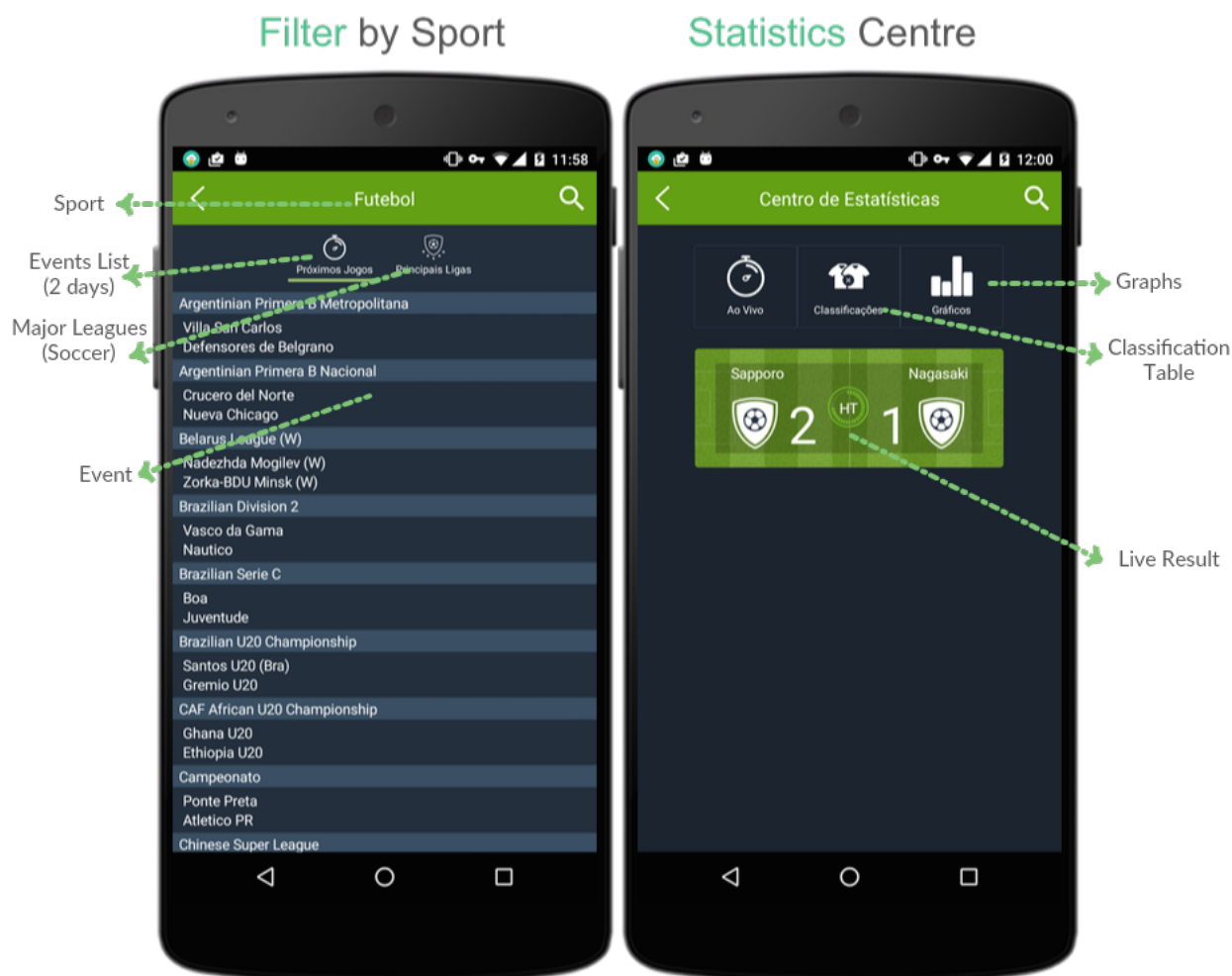


Figure 6.11: Filter by Sport and Statistic Centre Screenshots

One of the features of Traderline Mobile is the possibility to follow the results of the matches live, as well as some important statistics for the analysis of those matches. When we enter the Stats Centre, we can see (when available) the result of the match, as well as important information on the match, such as all the information related to cards, goals, goal scorers and substitutions.

Traderline's settings can be found in the menu at the top, in Traderline Mobile's Home. In the General Settings tab, we can customise a different number of settings:

- Include Australian Markets - Adds the Australian markets to the market list;
- API Request - In the API settings, we can define the number of times per second that Traderline Mobile will update its data. Traderline is pre-set to make 10 requests per second, which corresponds to the Normal mode. The intensive mode increases the number of requests per second, but will increase the battery consumption of the device as well;

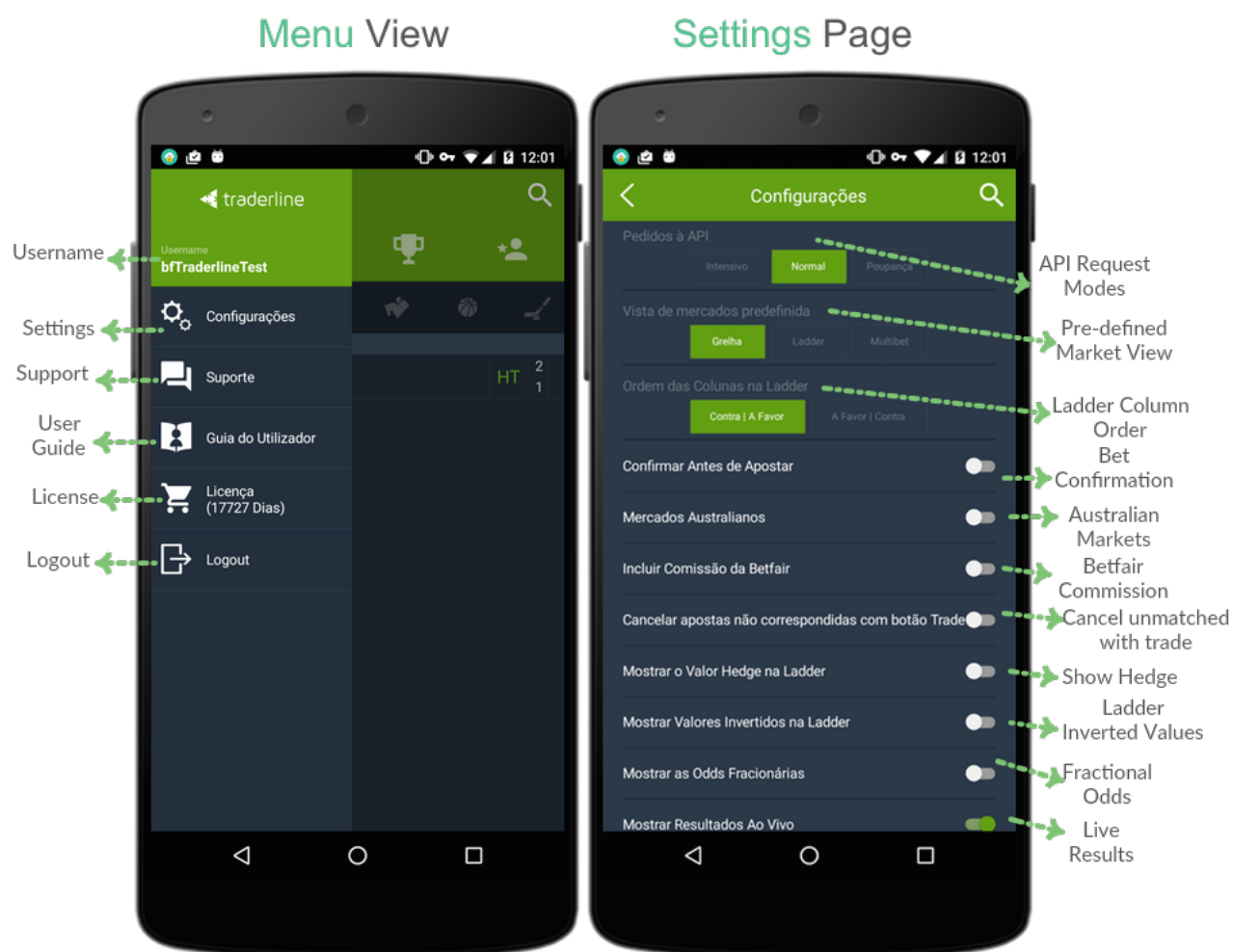


Figure 6.12: Menu View and Application Settings Screenshots

- Predefined Market Interface – We can define the interface that appears as soon as we enter the market of a certain event: Ladder, Grid or Multibet interface;
- Include Betfair Commission – We can configure Traderline Mobile to display our profit already without the amount that corresponds to the Betfair Commission. This way, we will always get our net profit;

- Cancel the Unmatched Bets with Trade – When placing a bet, in case that bet remains unmatched, we can configure Traderline Mobile to automatically cancel that bet when we touch the Trade button;
- Show the Hedge Value in the Ladder - Always shows the values at the left of the column, in hedge, in the ladder interface;
- Order of the Columns in the Ladder / Inverted Values in the Ladder – Change the order of the columns in the ladder interface;
- See Fractional Odds - See the market with fractional odds (typical in the UK) or decimal odds;
- Display Live Scores – The live scores feature can suffer some delays. It is possible to disable that service, so that Traderline Mobile does not display any information on the match other than that already provided by Betfair;
- Hide the Account Balance – In case we wish to hide our account balance;
- Enable Sound Alerts - Enable sound alerts for when the market is suspended, for when placing bets, before the market starts and when the market is about to go live.
- Edit the Main Sports – Edit the sports in the Main Sports menu, in Traderline Mobile's Home.

All these features are represented in figure 6.12.

6.4.2 Traderline Application progression

The old Traderline mobile version had several problems regarding performance, functionalities, design and, to sum up, quality. These were important points that were addressed in the beginning to project and were used when planning plan the development of the new application. The technology was an important point to dismiss some potential problems. Xamarin, and its shared project capabilities, allowed the development team, composed by the intern and other members, to work together, share the base codebase of the project and discover problems while developing for its different view layers (Mobile / Desktop). For instance, if a problem was discovered while building the mobile application and if it was a problem related to the *Core* module of the Traderline Solution, when correcting it, it would be fixed for all the different devices.

Table(6.2) summarises differences between the old Traderline Mobile application and the newly developed one. There are new features present, such as update system for the application, cashout confirmation, gestures, and bugs detected in the previous application were also fixed (Table 6.1).

Bug	Old Application
Favourite markets not being saved	YES
Multibet not working	YES
When entering Markets (Keyboard pop up)	YES
Stop loss not work ing	YES

Table 6.1: Bugs detected in the previous Traderline Application

One of the reasons for building a new Traderline Mobile application was the problems present in the previous versions, critical bugs, such as Multibet not working.

Feature	Old Application	New Application
Touch Gestures when accessing Menus	✗	✓
Update System	✗	✓
Cashout Confirmation	✗	✓

Table 6.2: Traderline versions comparison

There were new features implemented in the new Traderline Mobile solution. Adding gestures to the application allows a more native look and feel, using swipe to show or dismiss menus. The update system was also an important addition.

The Google Play Store¹, does not allow the publishing of gambling applications, and to avoid the users to be constantly installing and uninstalling the application, manually, it was developed a custom update system for Traderline Mobile. Every time the applications starts, it checks if it is the latest version released (REST API call to a service) and if there is a higher version available, it informs the user, downloads and updates the application. It is not possible to login Traderline unless the application is updated to the latest version.

¹Digital distribution service operated and developed by Google. It serves as the official app store for the Android operating system, allowing users to browse and download applications developed with the Android SDK and published through Google[37]

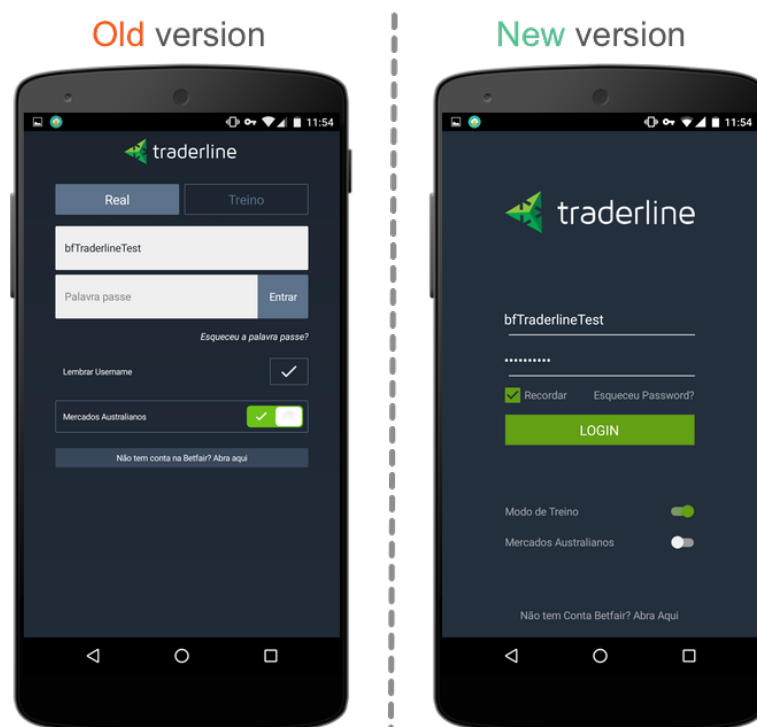


Figure 6.13: Previous Traderline Application and the New version

The Mobile development was faster than the implementation of the Desktop solutions, since its development team was also responsible for the *Core* module, so, while developing the Mobile application, the majority of the possible bugs would be discovered. The intern, along with the development team, would participate in this process to solve every problem so in a way, the responsibility was not just to the Mobile Application as a separate component, it was all part of the Traderline solution and required knowledge of the project as a whole.

The design was a responsibility of the company designers, however some suggestions were made in order to improve its usability which were gladly accepted and implemented.

By using Xamarin.Forms it was possible to implement the look and feel of mobile applications, with the different gestures (such as *swipe*) that provided a native touch.

The validation by the client was extremely important, as well as the testing with users, who discovered different problems that were properly fixed, improving the quality of the application before being released to the market.

In the final phase of the development of the Traderline Mobile Application, it was published a version of Tradeline Mobile for the Asian Markets, as result of a partnership between Stratio and another company.

This version of Traderline has only minimal design changes and, since it was released in May, still not contain some features that the final version will provide.

At the current moment, there is already more than 1000 people around the world using the application, with 251 active users, over than 22.3 million euros exchanged and more than 100 thousand transactions.

6.5 Testing

Traderline Mobile is a business solution for sports trading, dealing with extreme sensitive information as it is people's money. When a user installs Traderline Mobile, it expects to be completely functional in order to not lose any unexpected funds due to software problems. During the development of this application, specially in the second semester of the internship, it was released a version of Traderline Mobile to be distributed to a restrict number of alpha-testers[38]. From this process, a list of issues was reported, and then a period to fix all the problems was reserved, before continuing the development of the remaining features.

It might have been useful to do Usability Testing, however the applications screens were designed and validated by Stratio, and this type of tests were not planned for this internship.

6.5.1 Scripted Testing

Scripted testing was performed to ensure the mobile application developed meets the requirements for which it was designed. It is an approach that focus on precision and control, the tests are executed, well documented, progress towards 100% execution can be controlled and are easy to repeat when necessary. All the tests were designed by the intern, taking into consideration all the requirements defined in the first phase of the project, a sample of the structure of each test case is present in the table below (6.3)

Test Case ID	TC-LOGIN-01
Summary	User logs in the application
Preconditions	User has valid Betfair credentials
Step Actions:	Expected Result:
Fill Username and Password	Username and Password fields are displayed (password is protected with “*”)
Press Login button	Application navigates to Home Screen

Table 6.3: Scripted Test Case Scenario Example

All the tests where executed by the intern as well as the stakeholder, to ensure the validation of the implementation. It was performed a total of **40 test cases**, which where all validated with **success**. This type of testing is important, however it does not cover all the possibilities within a mobile application. There are many variables, and users never do what they are expected to do, to complement this, it was conducted exploratory testing. The use of beta-testers was fundamental in this process.

6.5.2 Exploratory Testing

Exploratory Testing involves simultaneously learning about the software under test while designing and executing tests, using feedback from the last test to inform the next[39]. Testers interact with the application in whatever way they want and use the information the application provides to react, change course, and generally explore the application's functionality without restraint. Advocates argue that exploratory testing allows the full power of the human brain to be brought to bear on finding bugs and verifying functionality without preconceived restrictions[40]. The most important phase of the exploratory

testing was conducted with a group of alpha-testers, 10 testers, 4 of them with knowledge of the previous Traderline Mobile Application.

This phase was extremely productive, since the testers provided feedback and input on several issues, the result of exploring the application, culminating in a total of **120 issues raised**, which where all properly fixed.

7

Conclusions

The following chapter concludes this report by addressing some thoughts on the work done, the lessons learnt during this internship, as well as some future plans.

It is fair to say that this internship was a success. It was, without doubt, an ambitious project. All the proposed objectives for this were reached:

- Learn the Business behind Trading: prior to this internship, the intern knowledge regarding sports betting, Trading, Betfair was very superficial. After this experience it was possible to understand how the Trading world works, though being such a complex area, there is still a lot to learn.
- Requirements Analysis - It was necessary to meet to with stakeholders to understand what was going to be developed in the Traderline Mobile solution, analyse the previous application, which is an important step of the engineering process.
- Technology: Using a Cross-Platform solution, in a complete new language was a challenge, however in the end it was possible to develop knowledge and skills regarding Xamarin, Xamarin.Forms, XAML, C#, Android. This is a technology that is clearly in expansion, being the future of mobile development, such knowledge will possible reveal to be extremely useful in the future.
- Integration in a Software Development Team: Communication is essential, this is a great lesson. The ability to watch and learn with other co-workers is also extremely helpful, it was an experience that allowed a more closer feel to the reality of a Software Engineer.
- Testing: Using real testers to test to the fullest the application was a great experience, there were a lot of minor mistakes that were only caught with heavy trading sessions. It was also a lesson, a product despite being ready, may not always be ready, the users try everything and a Software Engineer must predict great possibility of actions.

Building an application for a company, that has thousands of users, integrated in a development team, with completely new technologies was a challenge, but in the end all the difficulties were overcome. Certainly software engineering is an engineering discipline. The process of understanding specifications, architecture, implementation, testing requires dedication and will, and being part of a project from scratch, where all these skills were required and improved is definitely meaningful.

This internship also provided a very special contribution to the intern in several areas:

- Systems development life cycle: the process of software development, getting the right tools to do the job.
- Problem solving: developing software is all about problem solving. Writing code is only a tiny part of what a developer does - most of the work involves solving some really complex abstract problems, in this internship this was an essential skill necessary to develop the Traderline application.
- Communication skills: communicating with clients proved to be particularly tough, specially when having to explain highly technical concepts to non-technical people. It is important to know when to brief, and when to go into detail. A good communication, with clients or the team, can be the key to success, or in the other way, to failure.
- Effective estimations - estimating the time needed to execute a certain task is nowadays essential in Software Engineering. In the beginning, there is a tendency to be optimistic in the estimation process, however with practice, there are methods and habits that can be applied leading to more optimistic estimations.

After finishing the development process, there are some points that could be changed in order to improve the Traderline application. The inexperience using the Technology led to some mistakes that, today, would be done in a different way. Using Xamarin for the solution is no doubt a great investment, that pays off in terms of maintenance, quality when developing the different applications. However, the Xamarin.Forms for building the Traderline mobile application has its flaws, leading to a painful process when faced with complex User Interfaces as Traderline has.

Concerning future directions, there are some features that could be, in the future implemented in the Traderline Mobile application, one suggestion would be a recommendation system. Recommender Systems have become an important part of various applications, supporting both customers and providers in their decision making processes[41]. The aim of Recommender Systems is to suggest useful items to users, in the Traderline case, would be to suggest markets for the user to bet in, markets that the user would be potentially interested in. However, in recent years recommender systems concept has been extended in different areas like sport, [42][43]. By analysing the user profile, what markets he usually bets on, the type of sport he follows, he could be presented with alerts regarding important information. For instance, if a user bets on football events, specially in the Match Odds market, and usually backing a certain team, whenever there is a new event during the season involving that team he could be notified. We could go even further, by analysing the odds that the user has preference on and match all the information to make an even more informed suggestion.

This approach can also be generalised or transformed into a gamification[44] concept, promoting the continuous interaction of the user with the application.

This internship has been a very rewarding experience and, upon reaching the end, and looking back at all the successfully completed goals, the sense of accomplishment is remarkable. Being able to build a product from the ground up eventually releasing it to the market is a very enriching experience. The work in a real corporate environment is very demanding and inspired the intern to always try harder and do better to meet the clients expectations.

Bibliography

- [1] Dan Hermes. Mobile development using xamarin. In *Xamarin Mobile Application Development*, pages 1–8. Springer, 2015.
- [2] Xamarin. Xamarin.forms controls reference. <https://developer.xamarin.com/guides/xamarin-forms/controls/>, 2016. [Online; accessed 17-January-2016].
- [3] Stratio. Stratio. <http://stratio.pt/>, 2016. [Online; accessed 2-June-2016].
- [4] Wikipedia. Information and communications technology — wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Information_and_communications_technology&oldid=723246129, 2016. [Online; accessed 2-June-2016].
- [5] Wikipedia. Intelligent transportation system — wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Intelligent_transportation_system&oldid=723098553, 2016. [Online; accessed 2-June-2016].
- [6] Wikipedia. Industry 4.0 — wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Industry_4.0&oldid=722817194, 2016. [Online; accessed 2-June-2016].
- [7] Kent Beck, Mike Beedle, Arie Van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, et al. Manifesto for agile software development. 2001.
- [8] K Schwaber and Mike Beedle. *Scrum: the agile software development framework*. 2002.
- [9] Wikipedia. Trade — Wikipedia, the free encyclopedia. <https://en.wikipedia.org/wiki/Trade>, 2016. [Online; accessed 17-January-2016].
- [10] John Haigh. *Taking chances: winning with probability*. OUP Oxford, 1999.
- [11] Stephen J Hardiman, Peter Richmond, and Stefan Hutzler. Long-range correlations in an online betting exchange for a football tournament. *New Journal of Physics*, 12(10):105001, 2010.
- [12] Dominic Cortis. Expected values and variances in bookmaker payouts: A theoretical approach towards setting limits on odds. *The Journal of Prediction Markets*, 9(1):1–14, 2015.
- [13] Goal. Betting school: Understanding fractional & decimal betting odds. <http://www.goal.com/en/news/2994/betting/2011/01/10/2101368/betting-school-understanding-fractional-decimal-betting-odds>, 2011. [Online; accessed 17-January-2016].

- [14] Polyvios Tsirimpas. Specification and performance optimisation of real-time trading strategies for betting exchange platforms. 2014.
- [15] João Pedro Araujo Santos. A trading agent framework using plain strategies machine learning. 2014.
- [16] František Kopřiva. Constant bet size? don't bet on it! testing expected utility theory on betfair data. 2009.
- [17] Betfair. Betfair, internet betting exchange. <https://www.betfair.com/exchange>, 2016. [Online; accessed 31-May-2016].
- [18] BetOnDroid. Betondroid. <http://www.betondroid.com/en>, 2016. [Online; accessed 31-May-2016].
- [19] Rebel Betting. Rebel betting. <http://rebelbetting.com/>, 2016. [Online; accessed 31-May-2016].
- [20] Betfair. Betfair mobile. <http://www.betfair.com/android/>, 2016. [Online; accessed 31-May-2016].
- [21] Wikipedia. Android (operating system) — wikipedia, the free encyclopedia. [https://en.wikipedia.org/w/index.php?title=Android_\(operating_system\)&oldid=724038756](https://en.wikipedia.org/w/index.php?title=Android_(operating_system)&oldid=724038756), 2016. [Online; accessed 9-June-2016].
- [22] Betfair. What is dutching? <https://betting.betfair.com/what-is-dutching-betting.html>, 2016. [Online; accessed 31-May-2016].
- [23] Betfair. The art of bookmaking. <https://betting.betfair.com/the-art-of-bookmaking.html>, 2016. [Online; accessed 31-May-2016].
- [24] MobileBettingSites. Mobile betting. <http://www.mobilebettingsites.com/>, 2016. [Online; accessed 31-May-2016].
- [25] Jared Dickson. Xamarin mobile development. 2013.
- [26] Assist Prof Dr Volkan TUNALI. Comparison of popular cross-platform mobile application development tools. 2015.
- [27] Kevin Brennan et al. *A Guide to the Business Analysis Body of Knowledge*. Iiba, 2009.
- [28] Wikipedia. Extensibility — wikipedia, the free encyclopedia. <https://en.wikipedia.org/w/index.php?title=Extensibility&oldid=720180279>, 2016. [Online; accessed 1-June-2016].
- [29] Wikipedia. Usability — wikipedia, the free encyclopedia. <https://en.wikipedia.org/w/index.php?title=Usability&oldid=723061894>, 2016. [Online; accessed 1-June-2016].
- [30] APPDYNAMICS. What is mobile performance management. <https://www.appdynamics.com/info/what-is-mobile-performance-management>, 2016. [Online; accessed 2-June-2016].

- [31] Xamarin Inc. Introducing xamarin studio. <https://docs.xamarin.com/guides/cross-platform/xamarin-studio/>, 2016. [Online; accessed 7-June-2016].
- [32] Wikipedia. Git (software) — wikipedia, the free encyclopedia. [https://en.wikipedia.org/w/index.php?title=Git_\(software\)&oldid=723351634](https://en.wikipedia.org/w/index.php?title=Git_(software)&oldid=723351634), 2016. [Online; accessed 7-June-2016].
- [33] Anders Hejlsberg, Scott Wiltamuth, and Peter Golde. *C# language specification*. Addison-Wesley Longman Publishing Co., Inc., 2003.
- [34] Xamarin Inc. Xamarin.forms xaml basics. <https://developer.xamarin.com/guides/xamarin-forms/xaml/xaml-basics/>, 2016. [Online; accessed 7-June-2016].
- [35] Chris Newman. *SQLite (Developer’s Library)*. Sams, 2004.
- [36] Simon Brown. *Software Architecture for Developers*. Leanpub, 2015.
- [37] Wikipedia. Google play — wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Google_Play&oldid=726597000, 2016. [Online; accessed 23-June-2016].
- [38] Erik Van Veenendaal. Standard glossary of terms used in software testing. *International Software Testing Qualifications Board*, pages 8–9, 2012.
- [39] Juha Itkonen and Kristian Rautiainen. Exploratory testing: a multiple case study. In *Empirical Software Engineering, 2005. 2005 International Symposium on*, pages 10–pp. IEEE, 2005.
- [40] James A Whittaker. *Exploratory software testing: tips, tricks, tours, and techniques to guide test design*. Pearson Education, 2009.
- [41] Anna Gatzzioura and Miquel Sànchez-Marrè. A case-based recommendation approach for market basket data. *IEEE Intelligent Systems*, 30(1):20–27, 2015.
- [42] Pedro Henriques Abreu, Daniel Castro Silva, Fernando Almeida, and João Mendes-Moreira. Improving a simulated soccer team’s performance through a memory-based collaborative filtering approach. *Applied Soft Computing*, 23:180–193, 2014.
- [43] Pedro Henriques Abreu, Daniel Castro Silva, João Portela, João Mendes-Moreira, and Luís Paulo Reis. Using model-based collaborative filtering techniques to recommend the expected best strategy to defeat a simulated soccer opponent. *Intelligent Data Analysis*, 18(5):973–991, 2014.
- [44] Sebastian Deterding, Miguel Sicart, Lennart Nacke, Kenton O’Hara, and Dan Dixon. Gamification. using game-design elements in non-gaming contexts. In *CHI’11 Extended Abstracts on Human Factors in Computing Systems*, pages 2425–2428. ACM, 2011.
- [45] DI FH Norbert Haberl. Cross platform development possibilities and drawbacks of the xamarin platform. 2015.
- [46] Goal. Waterfall software development model odds. <http://www.oxagile.com/company/blog/the-waterfall-model>, 2014. [Online; accessed 18-January-2016].

- [47] Wikipedia. Internet of things — wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Internet_of_things&oldid=723173961, 2016. [Online; accessed 2-June-2016].

Appendix A

Flow Diagrams

A.1 Login

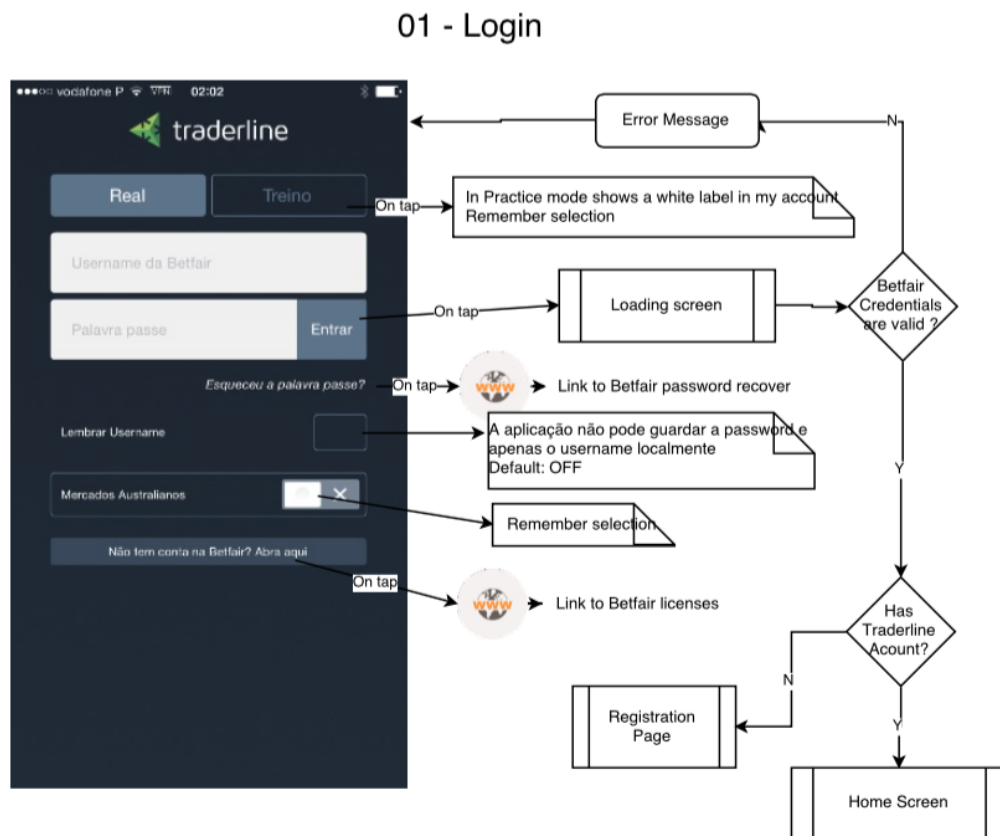


Figure A.1: Login Page Diagram

A.2 Home

02 - Home Screen



Figure A.2: Home Screen Page Diagram

A.3 Settings

03 - Application Settings



Figure A.3: Application Settings Page Diagram

A.4 Search

04 - Search Screen

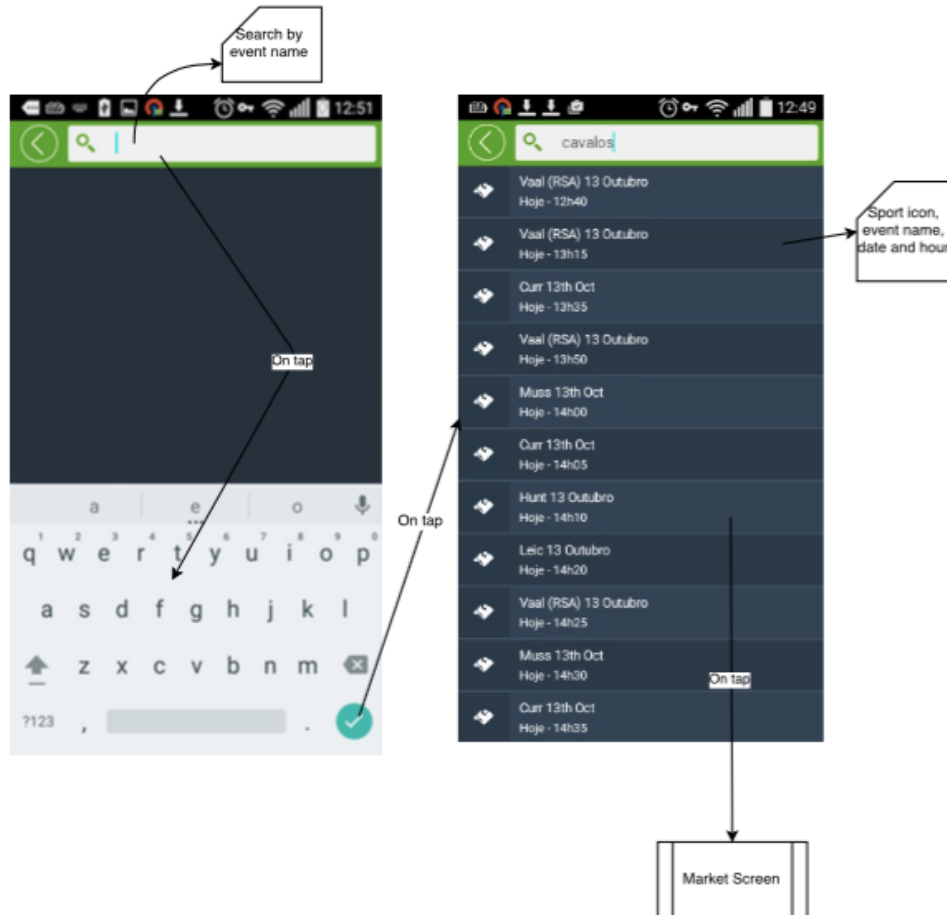


Figure A.4: Search Page Diagram

A.5 Live

05 - Live Events

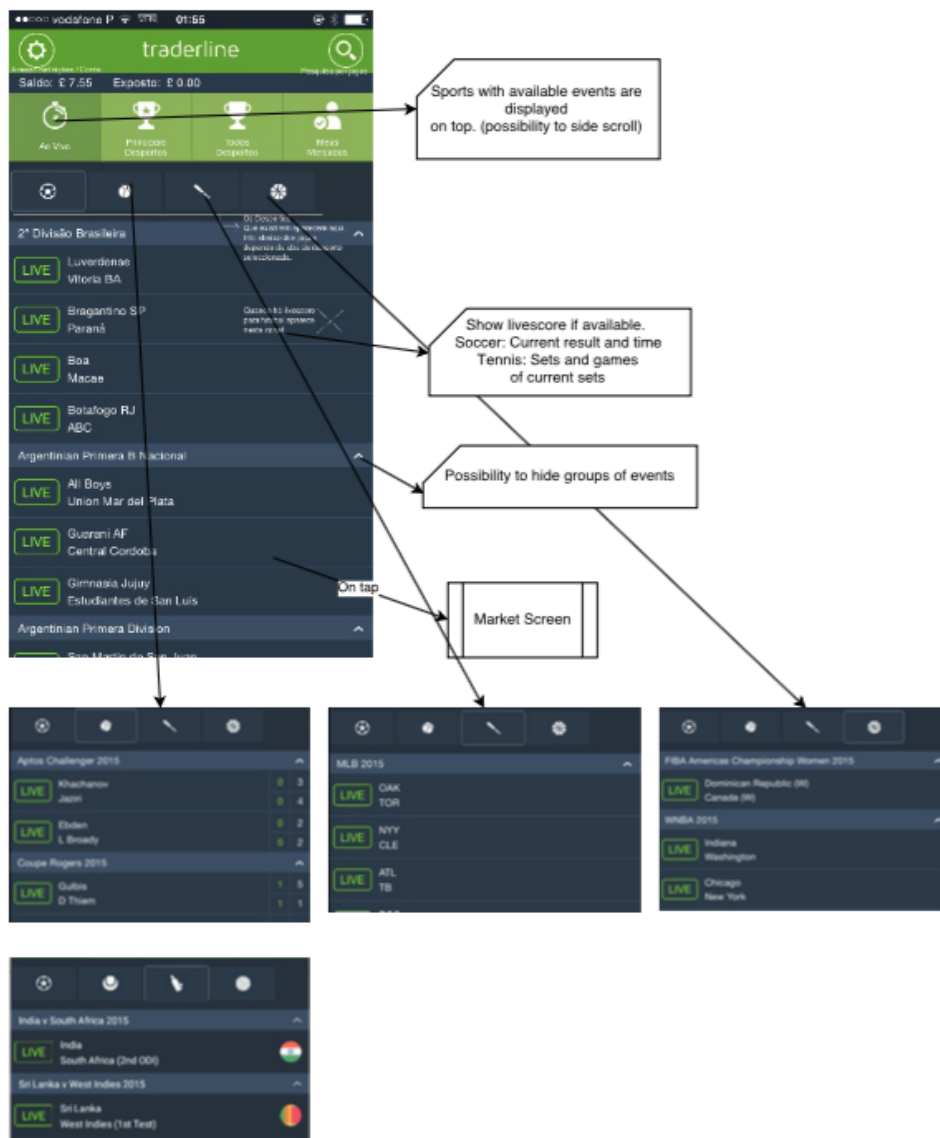


Figure A.5: Live Page Diagram

A.6 Favourite Sports

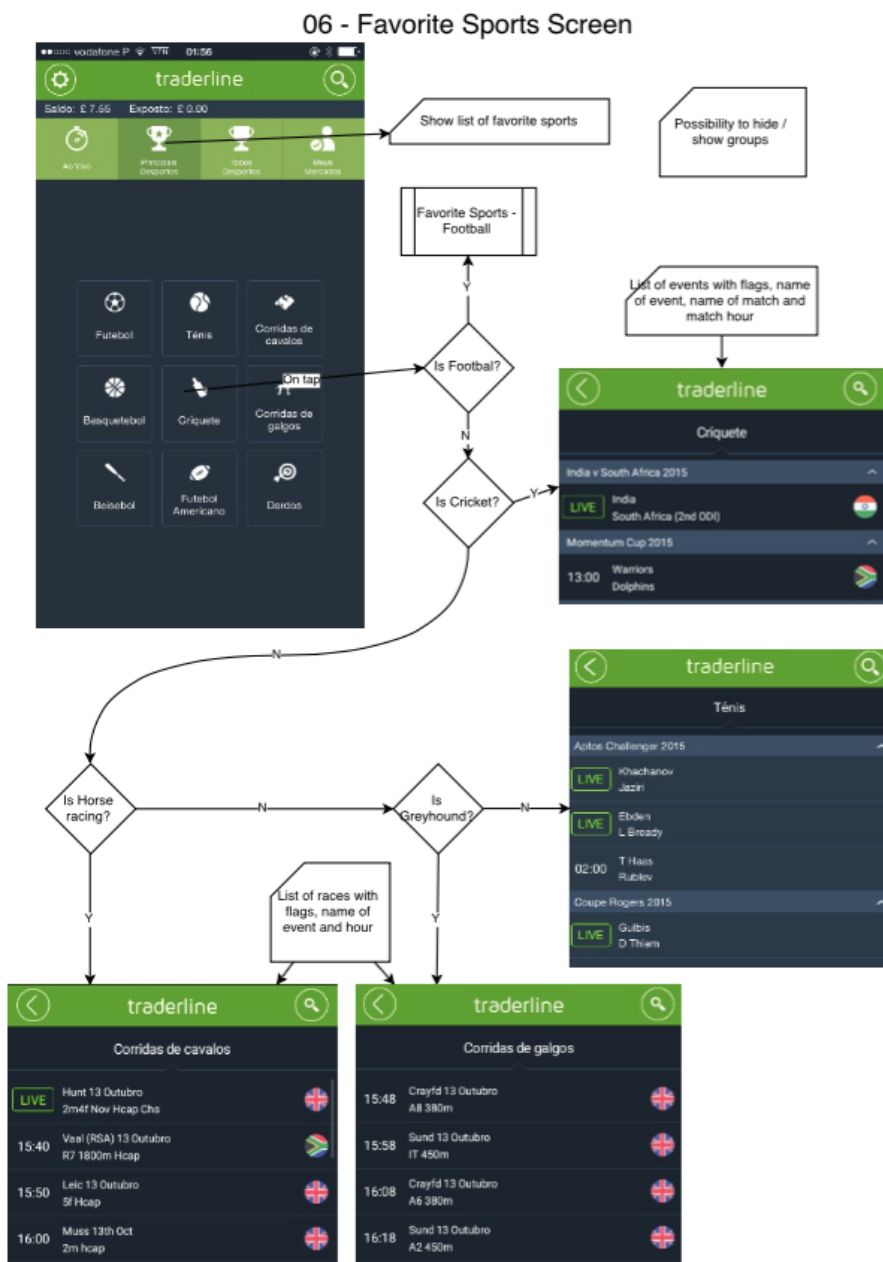


Figure A.6: Favourite Sports Page Diagram

A.7 Favourite Sports - Football

07 - Favorite Sports - Football

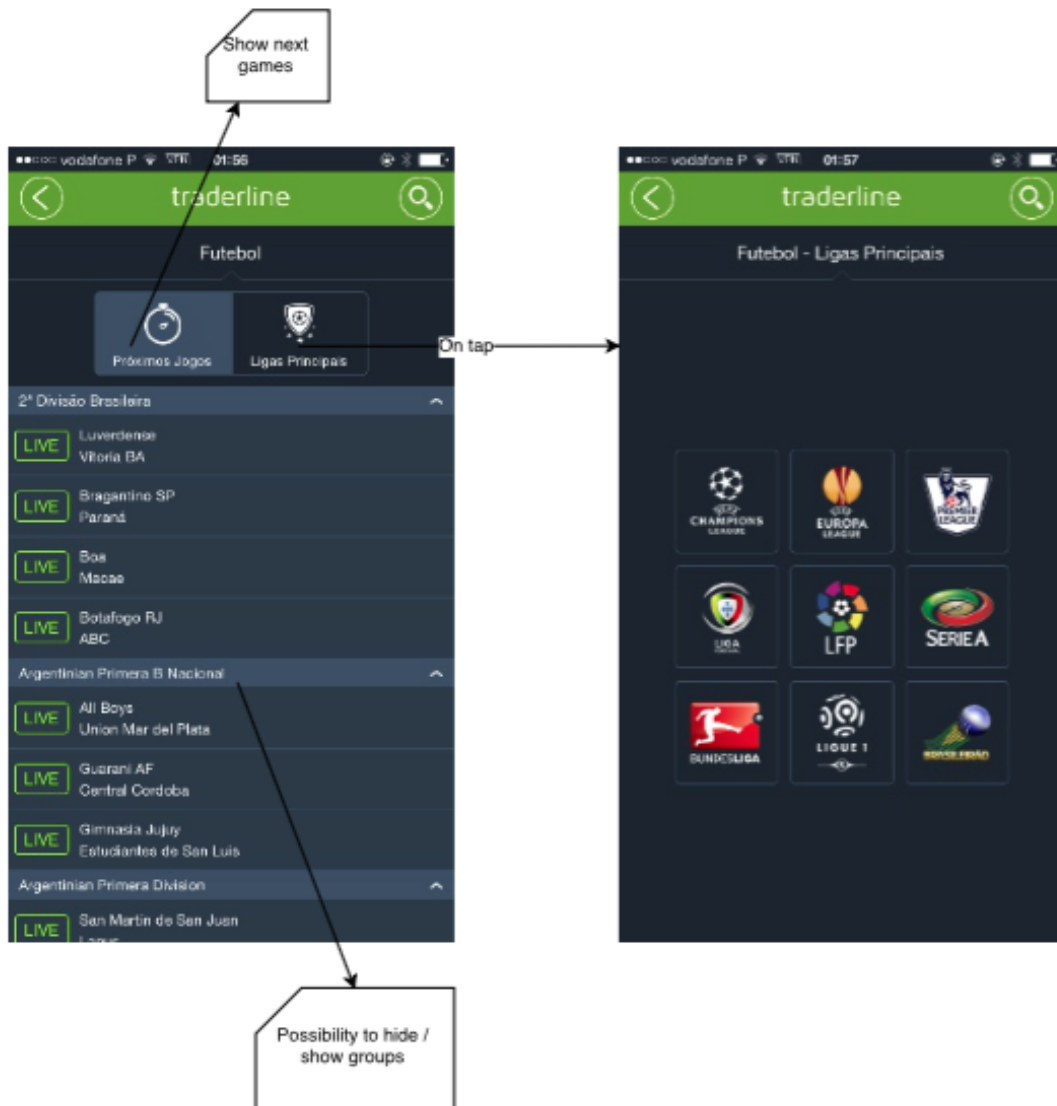


Figure A.7: Favourite Sports - Football Diagram

A.8 All Sports



Figure A.8: All Sports Diagram

A.9 General Settings



Figure A.9: General Settings Diagram

A.10 License

10 - License Expiration Date

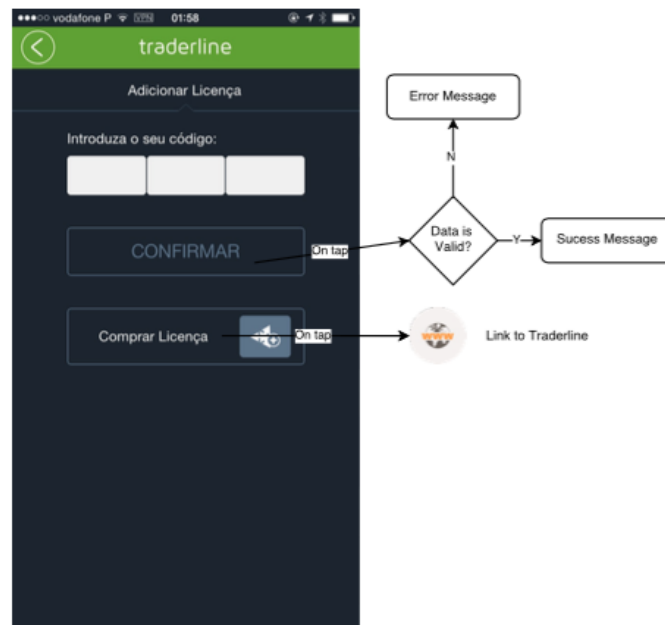


Figure A.10: License Diagram

A.11 Main Sports

11 - Configure Main Sports

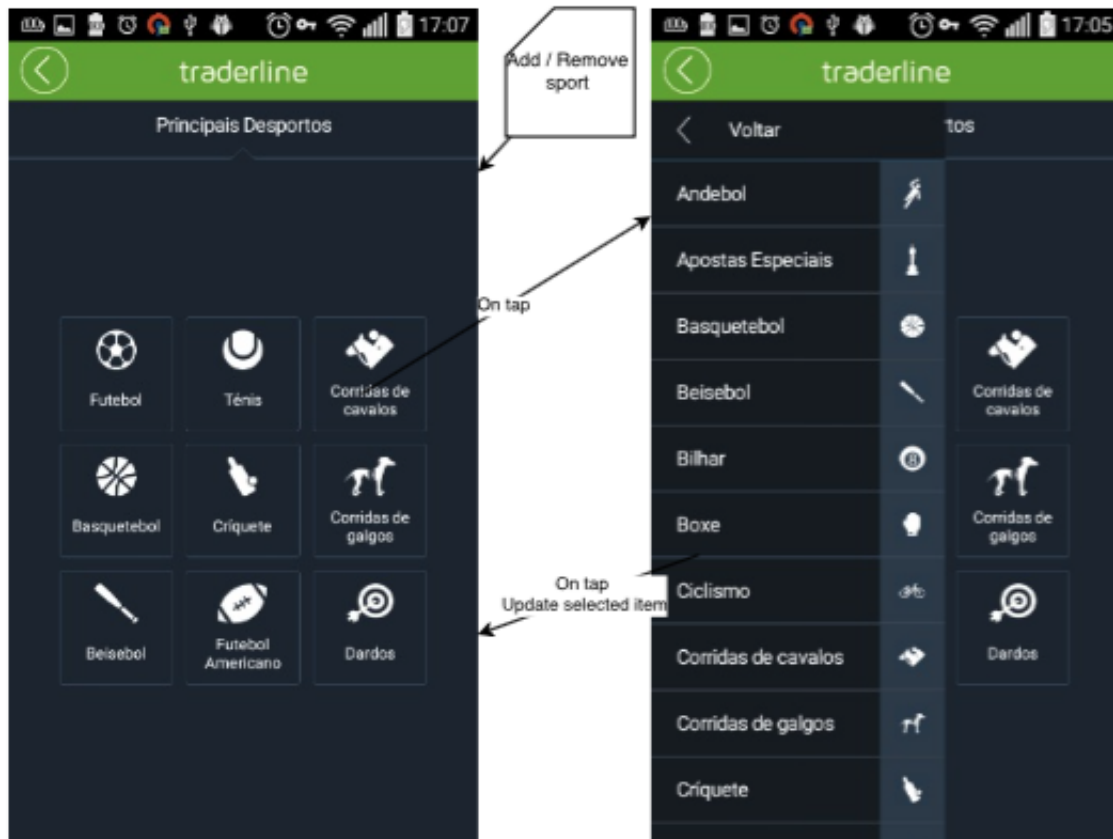


Figure A.11: Main Sports Diagram

A.12 My Markets

12 - My Markets

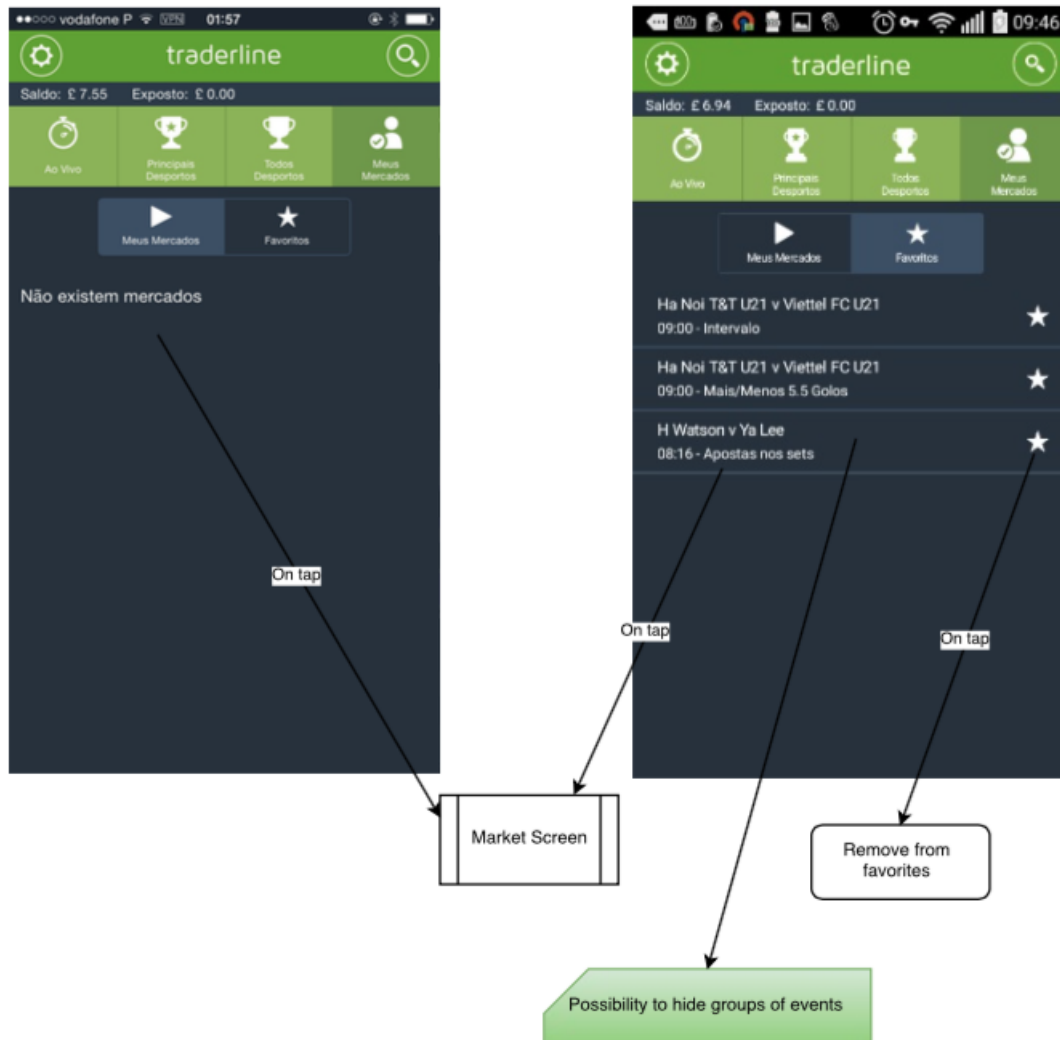


Figure A.12: My Markets Diagram

A.13 Market

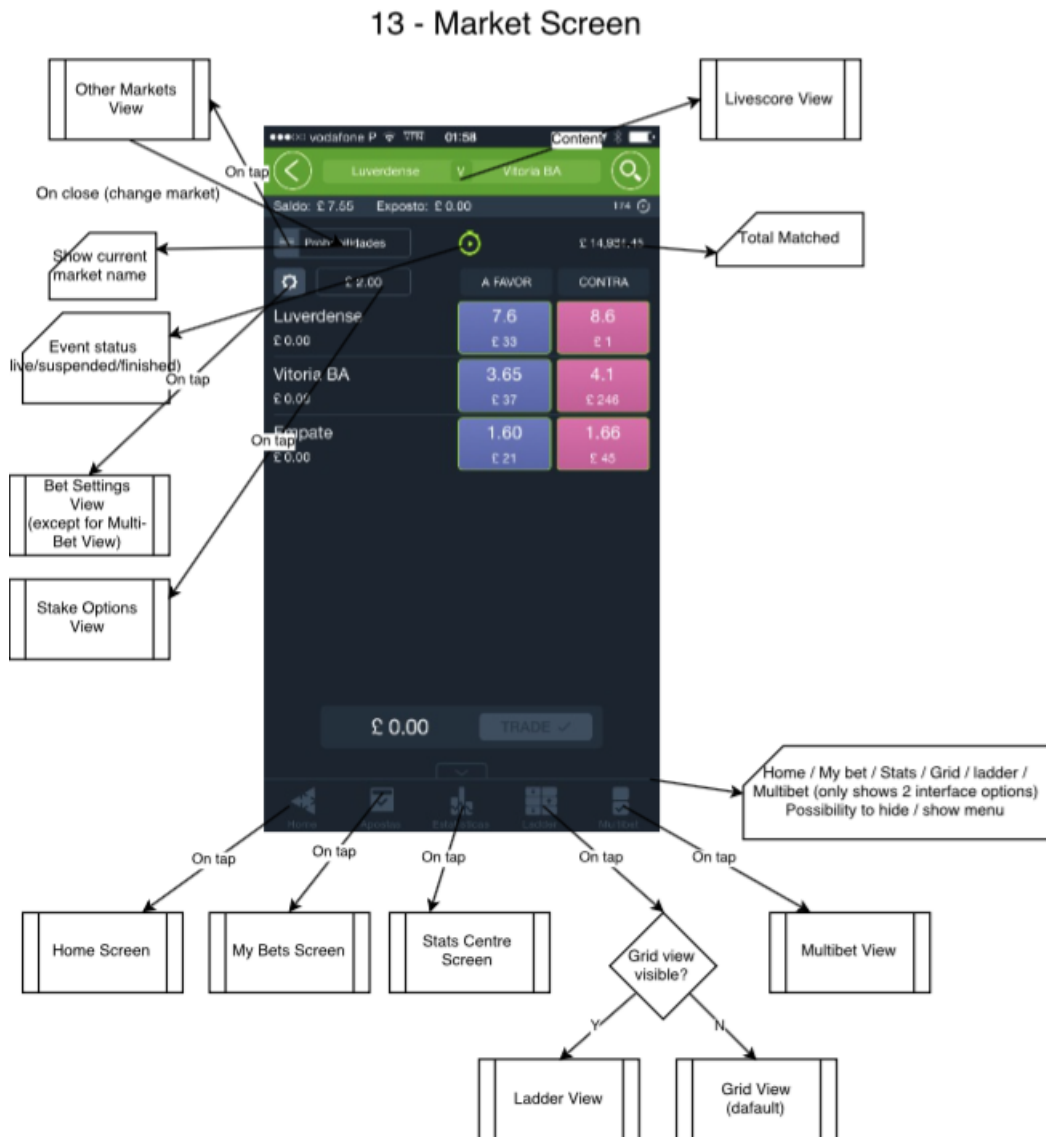


Figure A.13: Market Screen Diagram

A.14 My Bets

14 - My Bets Screen



Figure A.14: My Bets Diagram

A.15 Grid

15 - Grid View



Figure A.15: Grid View Diagram

A.16 Bet Slip



Figure A.16: Bet Slip view Diagram

A.17 Bet Settings

17 - Bet Settings View

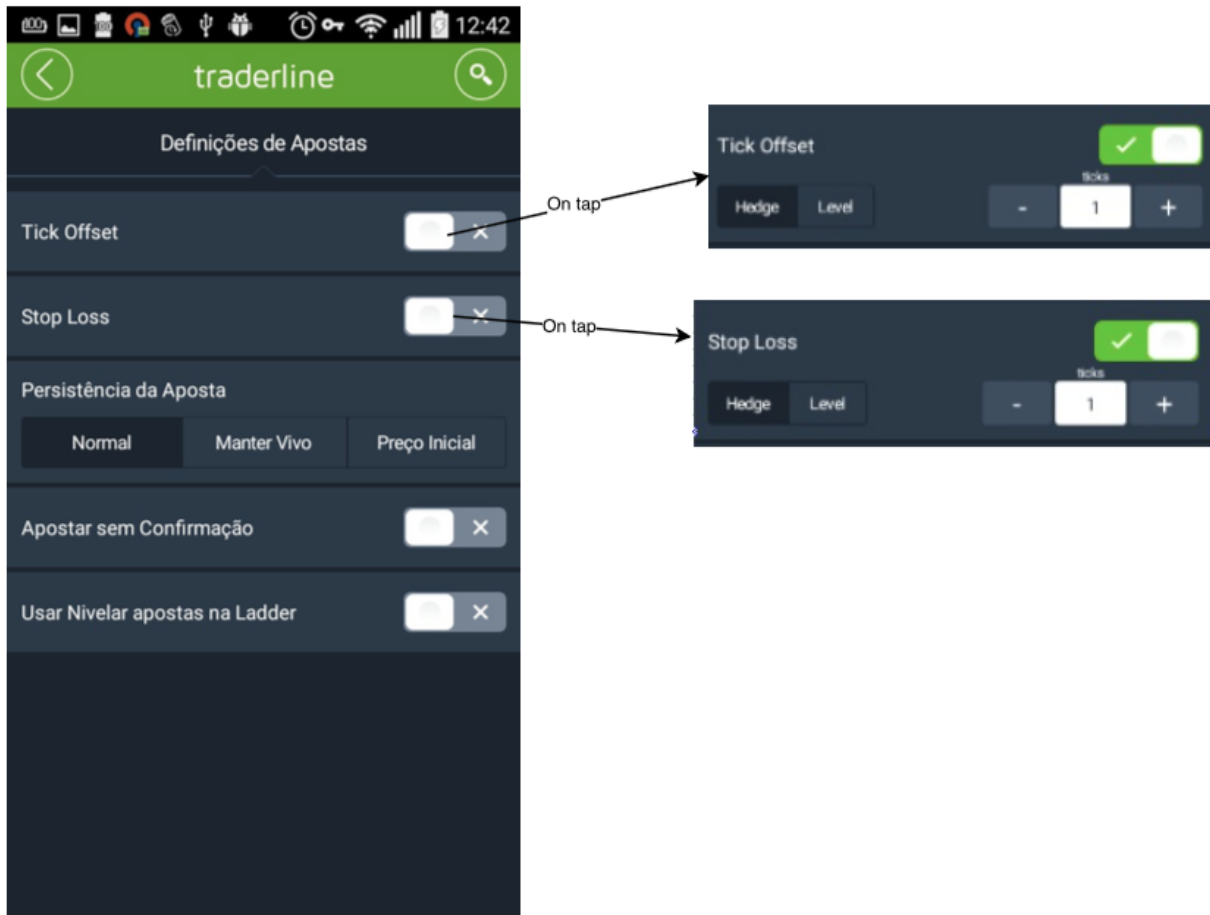


Figure A.17: Bet Settings Diagram

A.18 Other Markets

18 - Other Markets View



Figure A.18: Other Markets View Diagram

A.19 Stake Options

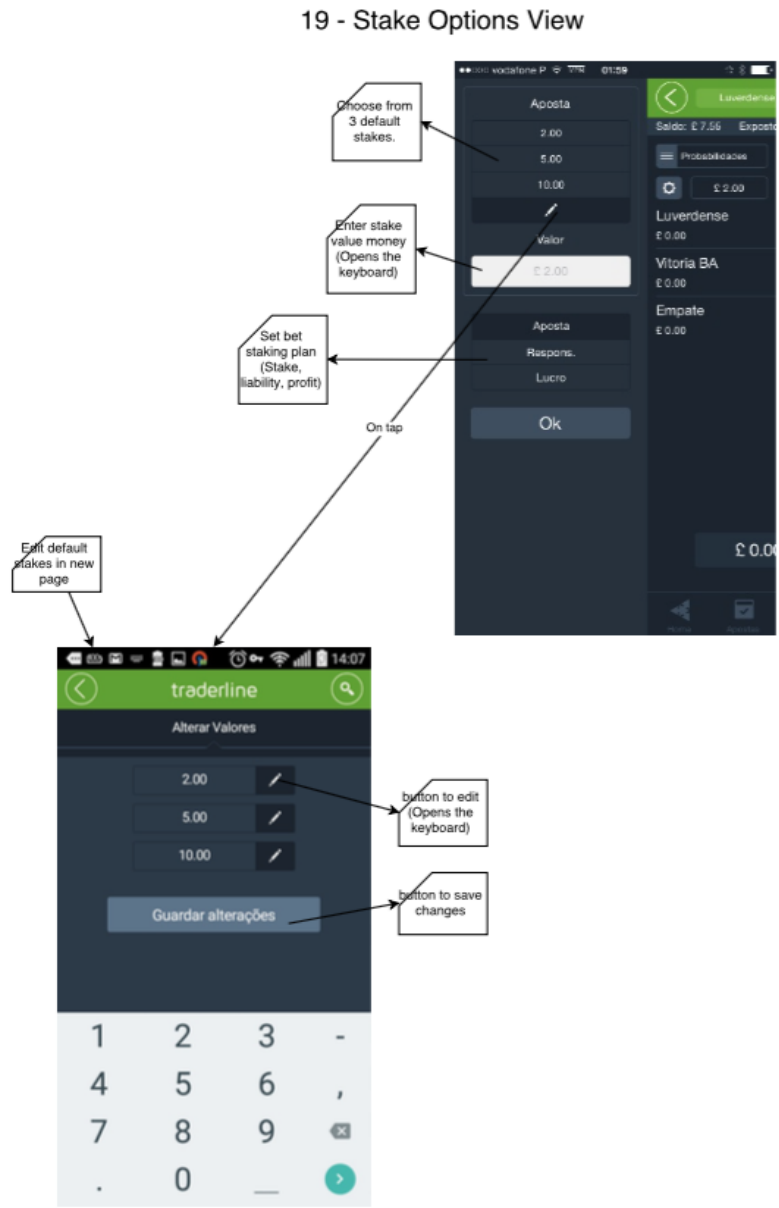


Figure A.19: Stake Options View Diagram

A.20 Ladder

20 - Ladder View



Figure A.20: Ladder View Diagram

A.21 Ladder Runners

21 - Change Ladder

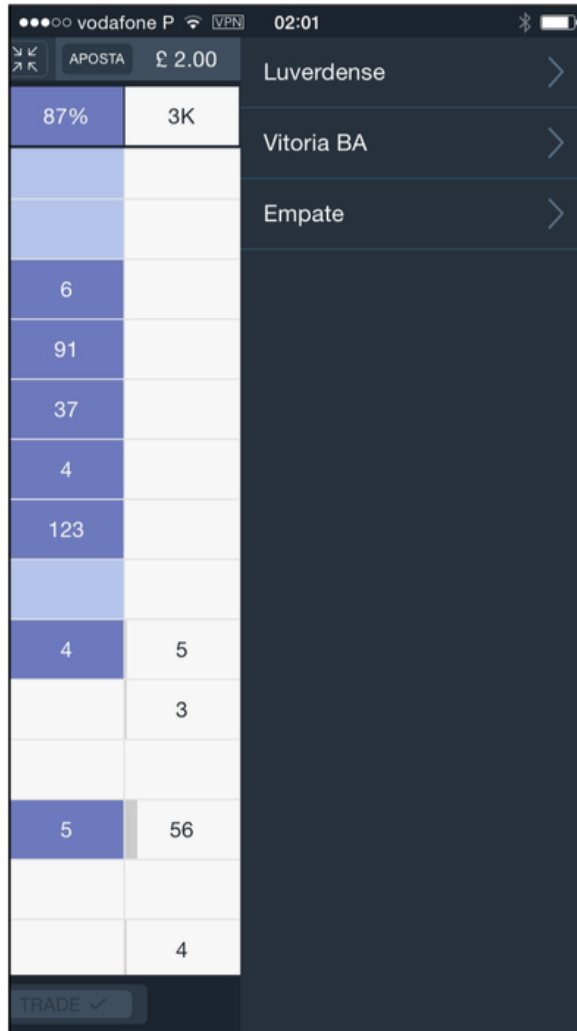


Figure A.21: Ladder Runners Diagram

A.22 Multibet

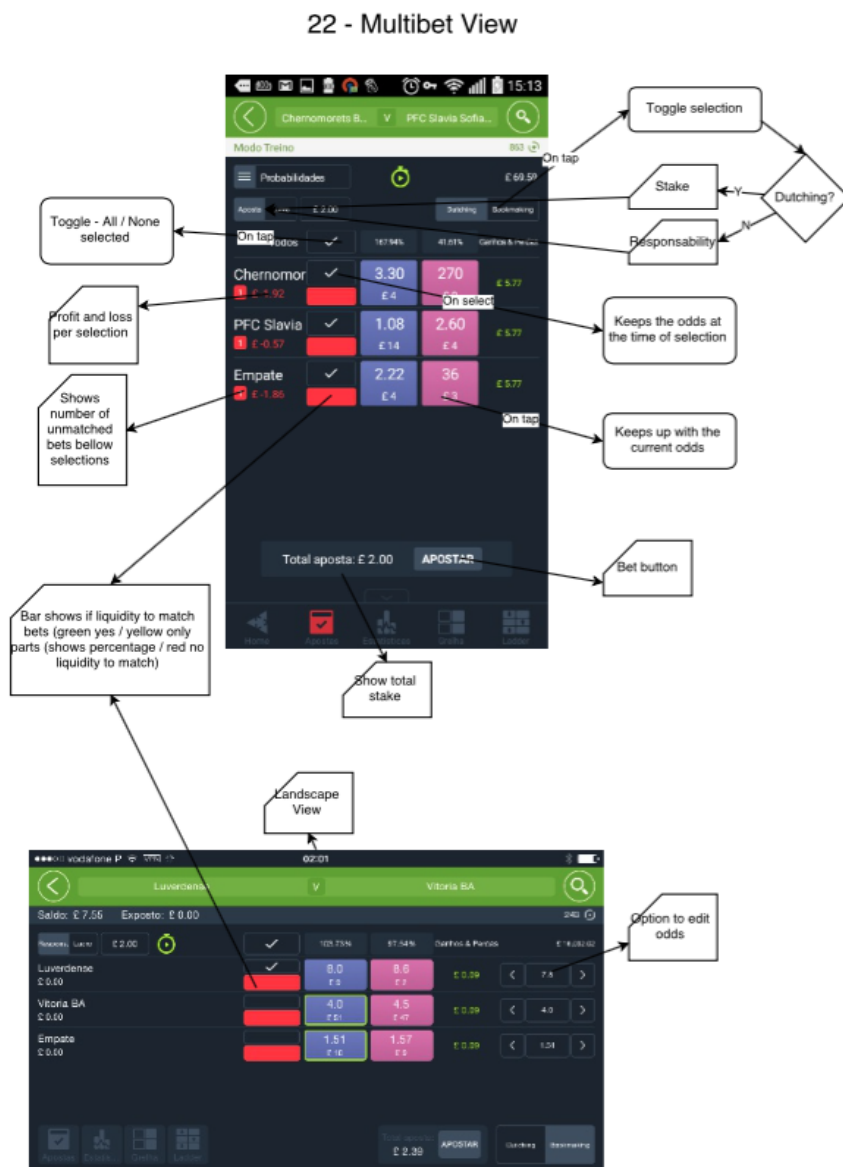


Figure A.22: Multibet View Diagram

A.23 Stats Center

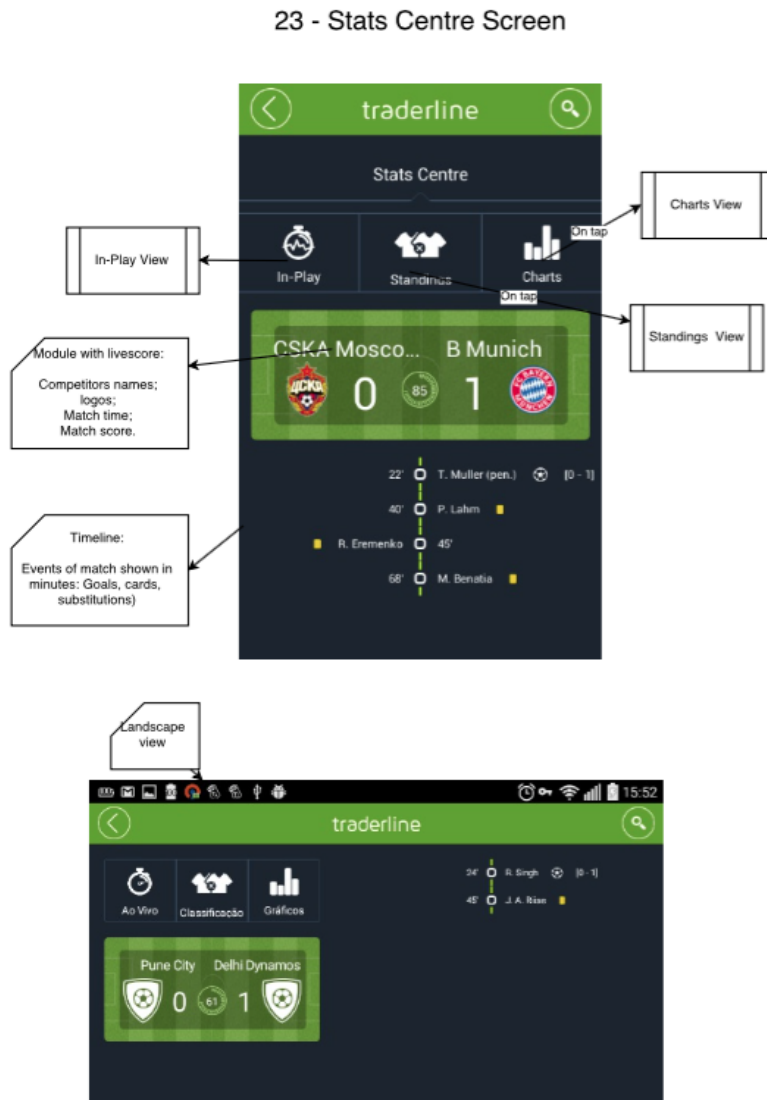


Figure A.23: Stats Center Diagram

A.24 In Play

24 - In Play View

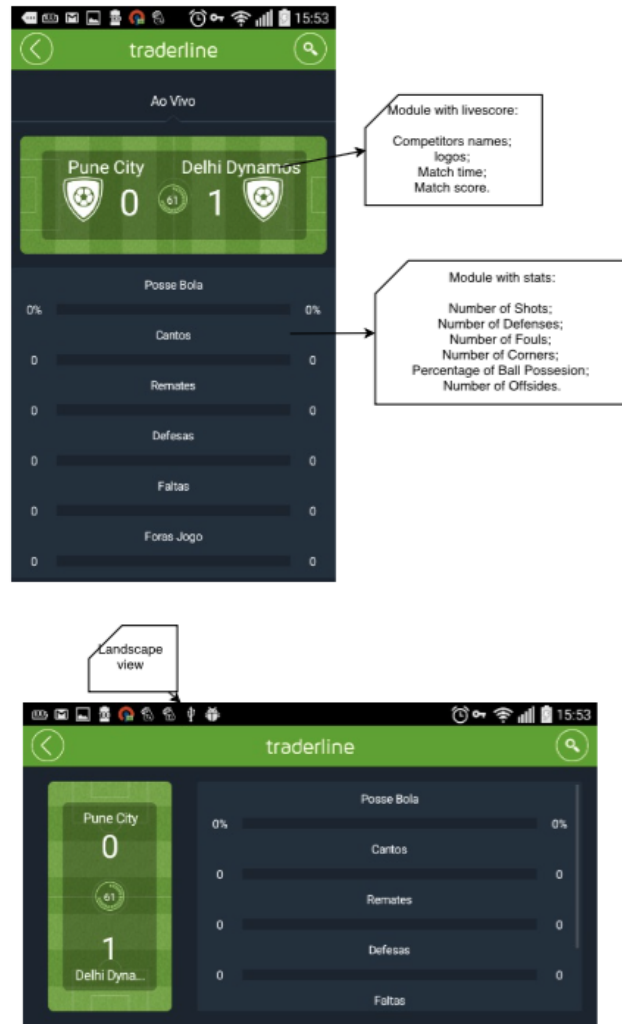


Figure A.24: In-Play view Diagram

A.25 Charts

25 - Charts View



Figure A.25: Charts view Diagram

A.26 Standings

26 - Standings View

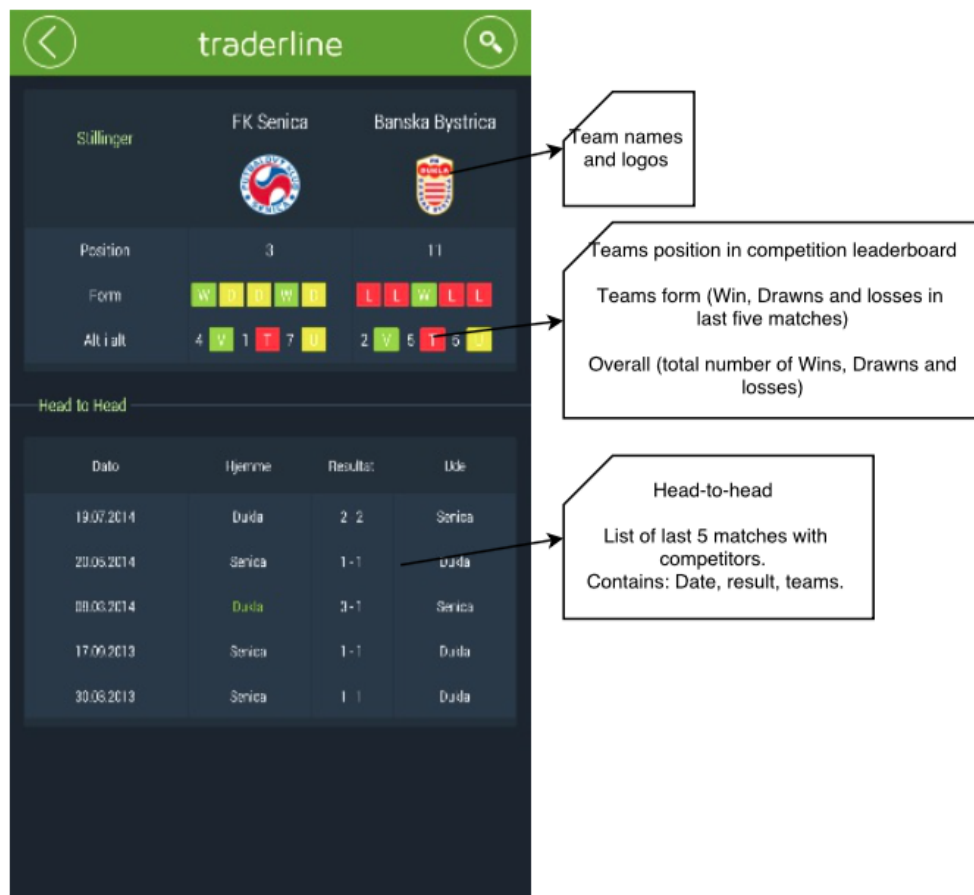


Figure A.26: Standings view Diagram

A.27 Live Score

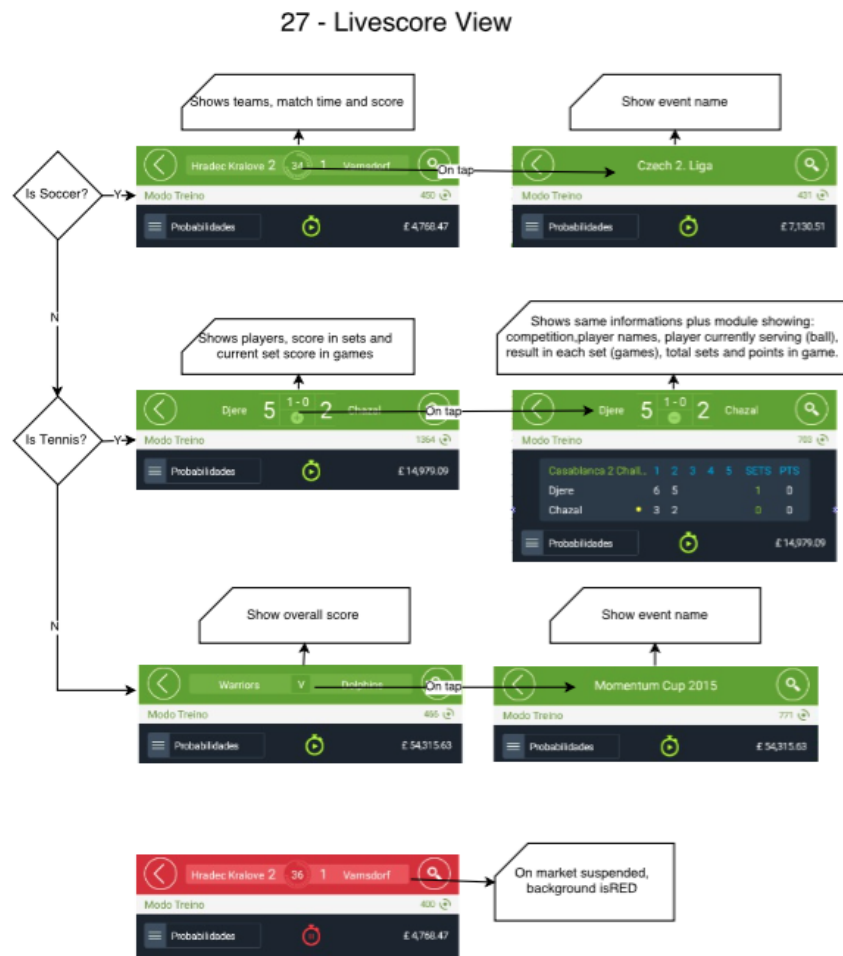


Figure A.27: Live Score view Diagram

Appendix B

Test Scripts

Test Case ID	TC-LOGIN-01
Summary	User logs in the application
Preconditions	User has valid Betfair credentials
Step Actions:	Expected Result:
Fill Username and Password	Username and Password fields are displayed (password is protected with “*”)
Press Login button	Application navigates to Home Screen

Table B.1: TC-LOGIN-01: Login in the application

Test Case ID	TC-LOGIN-02
Summary	User enables or disables “Remember Me”
Preconditions	User has valid Betfair credentials
Step Actions:	Expected Result:
Fill Username and Password	Username and Password fields are displayed (password is protected with “*”)
Enable Remember Me option	Remember Me box is activated
Press Login button	Application navigates to Home Screen
Terminate Application	Application is closed
Open Application	The previously inserted Username is displayed

Table B.2: TC-LOGIN-02: Remember me

Test Case ID	TC-LOGIN-03
Summary	User enables or disables “Practice Mode”
Preconditions	User has valid Betfair credentials
Step Actions:	Expected Result:
Fill Username and Password	Username and Password fields are displayed (password is protected with “*”)
Enable Practice Mode option	Practice Mode box is activated
Press Login button	Application navigates to Home Screen with practice mode enabled (No Account Balance is shown)

Table B.3: TC-LOGIN-03: Practice Mode

Test Case ID	TC-LOGIN-04
Summary	User enables or disables “Australian Markets”
Preconditions	User has valid Betfair credentials
Step Actions:	Expected Result:
Fill Username and Password	Username and Password fields are displayed (password is protected with “*”)
Enable Australian Markets option	Australian Markets box is activated
Press Login button	Application navigates to Home Screen with Australian Markets enabled

Table B.4: TC-LOGIN-04: Practice Mode

Test Case ID	TC-LOGIN-05
Summary	User makes a new Betfair account
Preconditions	User has no betfair account
Step Actions:	Expected Result:
Press Register account link	A new browser window appears linking to Betfair registration page

Table B.5: TC-LOGIN-05: Register Betfair account

Test Case ID	TC-LOGIN-06
Summary	User retrieves Betfair account password
Preconditions	
Step Actions:	Expected Result:
Press Forgot Password link	A new browser window appears linking to Betfair Recover Password page

Table B.6: TC-LOGIN-06: Retrieve account password

Test Case ID	TC-FAVSPORT-01
Summary	User access his favourite sports
Preconditions	User is logged in and in Home Screen
Step Actions:	Expected Result:
Press Favourite Sports Icon tab in Home Screen	A grid with the user favourite sports is shown

Table B.7: TC-FAVSPORT-01: Access Favourite Sports

Test Case ID	TC-FAVSPORT-02
Summary	User filter events by favourite sport
Preconditions	User is logged in and in Favourite Sports View
Step Actions:	Expected Result:
Press a sport Icon	New screen with a list of events filtered by the chosen sport

Table B.8: TC-FAVSPORT-02: Filter Favourite Sport Events

Test Case ID	TC-GENOPT-01
Summary	User enables Australian Markets
Preconditions	User is logged in and in General Options Page
Step Actions:	Expected Result:
Enable Australian Markets option	After returning to Live Events, Australian Markets will be shown, if available

Table B.9: TC-GENOPT-01: Australian Markets

Test Case ID	TC-GENOPT-02
Summary	User changes pre-defined Market View
Preconditions	User is logged in and in General Options Page
Step Actions:	Expected Result:
Choose a Market View Mode (Grid, Ladder, Multibet)	Selected Option will be highlighted
Open a new event	The default market view is the previously defined

Table B.10: TC-GENOPT-02: Market View Mode

Test Case ID	TC-GENOPT-03
Summary	User enables cancel unmatched bet with Trade
Preconditions	User is logged in and in General Options Page. User has unmatched bets in a Market
Step Actions:	Expected Result:
Enable cancel unmatched bets	Selected Option will be highlighted
Open the market with unmatched bets	A market page for the selected event will be opened
Press Trade Button	All the previous unmatched bets will be removed

Table B.11: TC-GENOPT-03: Cancel unmatched bets

Test Case ID	TC-GENOPT-05
Summary	User enables show hedge value in ladder
Preconditions	User is logged in and in General Options Page. User has open bets in a market
Step Actions:	Expected Result:
Enable show hedge value in ladder	Selected Option will be highlighted
Open the market with open bets in Ladder View	A market page for the selected event will be opened in Ladder View with hedge value in the first column

Table B.12: TC-GENOPT-05: Hedge value in Ladder

Test Case ID	TC-GENOPT-06
Summary	User changes Ladder column order
Preconditions	User is logged in and in General Options Page.
Step Actions:	Expected Result:
Select Ladder order:Lay-Back	Selected Option will be highlighted
Open the market in Ladder View	A market page for the selected event will be opened in Ladder View being Lay column the second column, and back column the fourth

Table B.13: TC-GENOPT-06: Ladder column order

Test Case ID	TC-GENOPT-07
Summary	User enables inverted values in Ladder
Preconditions	User is logged in and in General Options Page.
Step Actions:	Expected Result:
Enable show inverted values in ladder	Selected Option will be highlighted
Open the market in Ladder View	A market page for the selected event will be opened in Ladder View with inverted values

Table B.14: TC-GENOPT-07: Inverted values in Ladder

Test Case ID	TC-GENOPT-08
Summary	User enables fractional odds
Preconditions	User is logged in and in General Options Page.
Step Actions:	Expected Result:
Enable show fractional odds	Selected Option will be highlighted
Open the market	A market page for the selected event will be opened displaying odds in fractional mode

Table B.15: TC-GENOPT-08: Inverted values in Ladder

Test Case ID	TC-GENOPT-09
Summary	User enables show live results
Preconditions	User is logged in and in General Options Page.
Step Actions:	Expected Result:
Enable show live results	Selected Option will be highlighted
Return to homepage	Home page will be opened displaying live results for soccer and tennis events (if available)

Table B.16: TC-GENOPT-09: Live Results

Test Case ID	TC-GENOPT-12
Summary	User enables hide account balance
Preconditions	User is logged in and in General Options Page.
Step Actions:	Expected Result:
Enable show account balance	Selected Option will be highlighted
Return to homepage	Home page will be opened and balance will be empty

Table B.17: TC-GENOPT-12: Account balance

Test Case ID	TC-GENOPT-13
Summary	User disables application sounds
Preconditions	User is logged in and in General Options Page.
Step Actions:	Expected Result:
Disable application sounds	Selected Option will be highlighted
Open market	Market page will be opened
Place bet	No sound will be playing after placing the bet

Table B.18: TC-GENOPT-13: Application sounds

Test Case ID	TC-GENOPT-14
Summary	User edits favourite sports
Preconditions	User is logged in and in General Options Page.
Step Actions:	Expected Result:
Tap manage favourite sports	List of favourite sports will be displayed
Tap a sport to edit	List of available sports will be displayed
Tap sport from the available sports	Sport will be added to favourites, replacing the previous one selected

Table B.19: TC-GENOPT-14: Favourite Sports

Test Case ID	TC-ALLSPORT-01
Summary	User navigates to all sports page
Preconditions	User is logged in and in Home Page.
Step Actions:	Expected Result:
Tap All Sports	List of all sports will be displayed
Tap a sport	List of available events will be displayed

Table B.20: TC-ALLSPORT-01

Test Case ID	TC-SEARCH-01
Summary	User navigates search view
Preconditions	User is logged in and in Home Page.
Step Actions:	Expected Result:
Tap search button	Search view will be displayed
Type a Team / Event to search	List of matching events will be displayed
Tap event	A new market page will be opened for the selected event

Table B.21: TC-SEARCH-01

Test Case ID	TC-MARKET-01
Summary	User navigates to a market page
Preconditions	User is logged in and in Home Page.
Step Actions:	Expected Result:
Tap live event	Market page for the selected event will be opened
Type a Team / Event to search	List of matching events will be displayed
Tap event	A new market page will be opened for the selected event
Tap My Bets	A new My Bets page will be opened for the selected event

Table B.22: TC-MARKET-01

Test Case ID	TC-STAKE-01
Summary	User edits stake
Preconditions	User is logged in and in Market Page.
Step Actions:	Expected Result:
Tap Stake	Stake Options view will be opened
Choose stake	Chosen stake will be the new default value for the selected market
Edit Bet Staking Plan	Chosen bet staking plan will be selected for the market

Table B.23: TC-STAKE-01

Test Case ID	TC-MYBETS-01
Summary	User consults My Bets
Preconditions	User is logged in and in Market Page.
Step Actions:	Expected Result:
Tap My Bets	My Bets page will be displayed, with a list of all matched bets and unmatched, if available
Choose cancel unmatched bets	All unmatched bets for the current market will be canceled
Edit Bet	Bet will be updated with the new values

Table B.24: TC-MYBETS-01