

Betgogo23

Technical Document



**UNIVERSITY OF
BIRMINGHAM**
JUNIOR SOLUTION

A project presented by Birmingham Innovation Studio

Date: 12/01/2002

1. Introduction

This document explains the functional specification of toto, the None project in which Birmingham Innovation Studio, engages to fulfill all the requests demanded by None as explained during elementary meetings and calls.

1.1 Purpose of this document

The functional specification document serves the same purpose as a contract. The specifications presented in this document are related to the conversations done by None. Following this document, the developers agree to provide the capabilities specified and fulfill all requests and demands of None, who also agreed to find the product satisfactory if it provides the capabilities as specified.

1.2 Scope

betgogo is a web app that allows user to place bet between each other, this app will have a login and sign up page where users can login or register, reset their password and reset their email. It will have a profile page where users can update their profile and manage their account. there will be a friend page where users can manage their relationship with others. there will be a bet page where users can see their current bet, create new bets and modify existing bets.

1.3 References

All meetings, emails and materials exchanges with None.

2 Requirements

2.1 Functional Requirements

1. Authentication Pages:

- The web app must allow users to sign in using email and password.
- The web app must allow users to sign up with a unique email and password.
- The web app must provide a 'Forgot Password' feature that sends a reset link to the user's email.
- The web app must offer an option for email verification during the sign-up process.

2. Profile Page:

- The web app must allow users to update their profile information.
- The web app should provide a section to display the user's betting history.
- The web app must provide an option to delete the user's account with a confirmation dialogue.

3. Friend Page:

- The web app must allow users to search for other users and send friend requests.
- The web app should offer a feature to manage and edit friend lists.
- The web app must notify users of friend requests and accept/decline options.

4. Bet Page:

- The web app must allow users to view their current bets.
- The web app should offer features to create new bets and modify existing bets.
- The web app must provide real-time updates on the status of each bet.

2.2 Non-Functional Requirements

1. Reliability:

- The web app must have a minimum uptime of 99.99%.
- The web app should have a failover mechanism to switch to a backup service in case of server failure.
- The web app should offer offline access to stored betting history and active bets.

2. Performance:

- The web app should load user profiles and betting history within 2 seconds.
- The web app should be optimized for low-latency data fetching for real-time bet status.

3. Usability:

- The web app must be user-friendly with an intuitive UI/UX design.
- The web app should offer a tutorial and FAQs for first-time users.

- The web app should be accessible, following WCAG 2.1 guidelines.

4. Security:

- All user data must be encrypted both in transit and at rest using AES-256 encryption.
- The web app must comply with data protection regulations, including not sharing user data with third parties without consent.
- The web app should have a secure authentication mechanism, including two-factor authentication.