SOFTWARE REQUIREMENTS SPECIFICATION

for

ALLERGY TRACKING APP

Version 1.3

Prepared by : Anna Chechenina
Yvette Dimitrova
Stefano Galiano

May 30, 2023

Contents

Version	Date	Changes	IDs
1.1	19/05/2023	Added more detailed description of non-functional requirements Added propositions on app design decisions; Added versions table; Reformatted requirements section with a table and IDs;	
1.2	26/05/2023	Added sequence diagram, use-case diagram, actions diagram;	
1. 3	28/05/2023		

1 Introduction

1.1 Purpose

The "Breath easy" is an Android and IOS app that will help people with allergies to keep track of allergens in the nearby area. The app will contain a map where users will see alerts from other people and will be aware of specific allergens.

This will be especially relevant for people with severe allergies to prepare for the possible allergens before symptoms will hit them.

1.2 Intended Audience and Reading Suggestions

"Breath easy" is for people with allergies, mainly air allergies such as pollen, dust, ticks, mites, and chemicals.

1.3 Project Scope

"Breath easy" will function as a network of people with similar allergies.

- Sign-in/sign-up options. The users will be able to sign up using different social networks and log in later to the created account. Once the user is registered, we will provide a quick-entry questionnaire on types of allergies to personalize the interface.
- There will be 3 main sections in the App: the main page with some highlighted information, a map section with a monitoring stations layer, and with users' pinpoint; and the tips and articles section with some tips from experts.
- Users will be able to view all sections, fill in everyday questionnaires and submit alerts.
- We also employ Health-tracking devices and app support that will be an additional source of health parameters and will be used for making alert suggestions.

2 Acronyms

3 Overall Description

3.1 Product Perspective

"Breathe easy" is a more dynamic way of monitoring coming waves of allergens in the densely populated areas. Previously, the only source of information was very poorly organized websites of the local monitoring stations, but with our app all information will be derived automatically and will be presented in the user-friendly format on the map. We also introduce a brand-new feature with user-to-user information sharing by submitting alerts.

3.2 User Classes and Characteristics

Overall, "Breathe Easy" has three types of users:

- Allergic user - user that has viewer permissions as well as alert-submitting permission
- Expert - have all allergic user permissions and options, plus can submit an expert article
- Developer, database coordinator, support team - their role is to maintain the functionalities of the app, provide quick responses to user inquiries, and maintain the database integrity of the map interface

3.3 Database

"Breathe easy" store all the information of the allergic user. Before using the main function of the software,allergens map, users have to be registered.
All users have - login_parameter, user_name, first_name, last_name, user_id, role, post, email, phone_number, present_address, parmanent_address, allergies, password_hash and timeStamp.
Users have some extra informations after complete his/her registration , such as - user_id(foreign key), registration_id, quizz_allergies, temperature_notification, daily_quizz(foreign key), forecast_alert(foreign key), user_alert(foreign key). These are the information that contains the firs time that the user logs in
Each programs has some data - program_name, program_id, course_id, nunmber_of_semester, total_credit and course_length. There will be onew or many course_id in each programs.

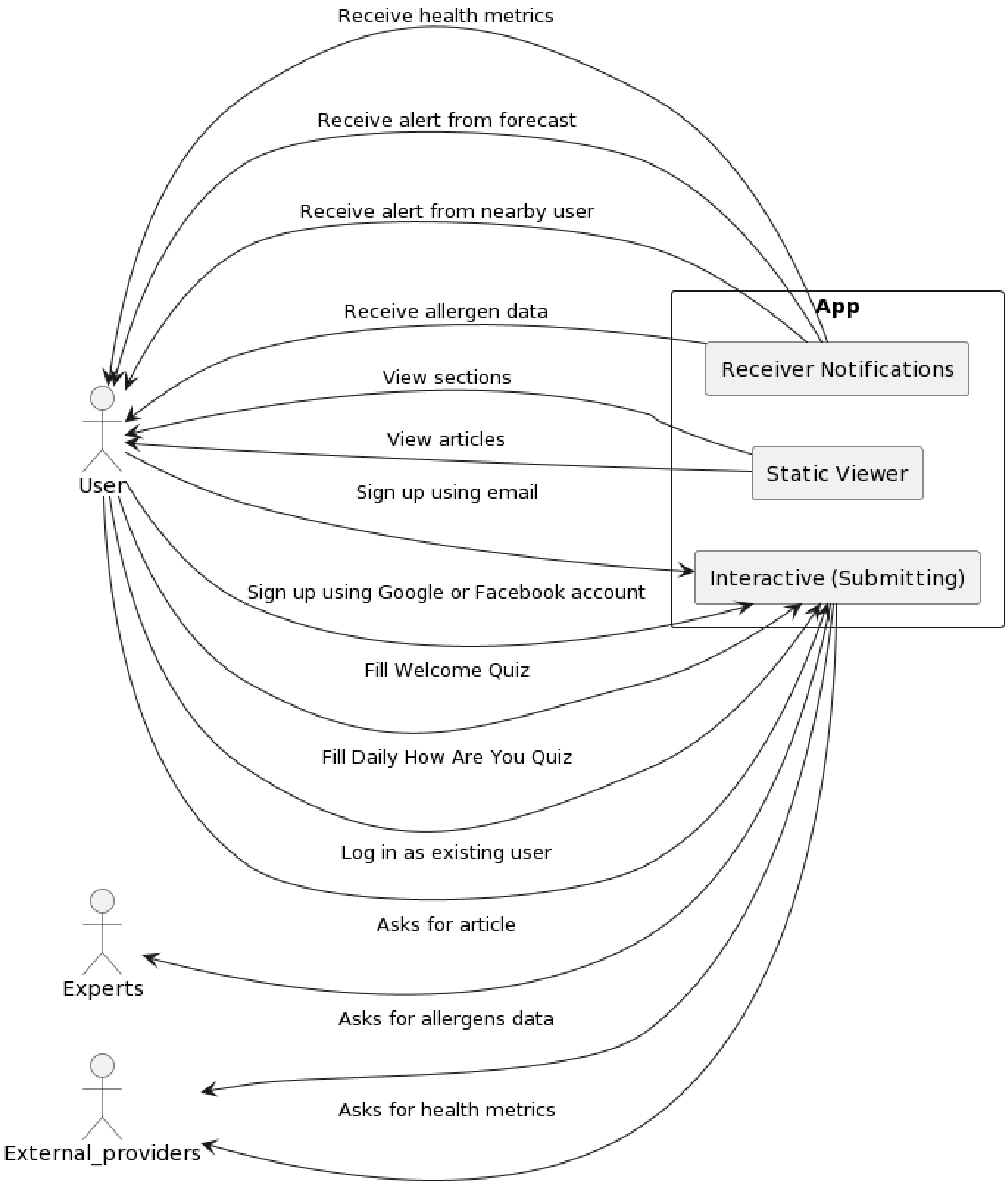

Figure 3.1: Data Flow Diagram

Courses table contains - course_name, course_id, course_code, credit, semester and teacher_id.

Every course has its own Credit Values. Those have been 2 types - lab, theory.
Result is the main feature of all. It contains the values of all the exams of a particular student. It has data field - student_id, course_id, term_test, attendance, marks(A), marks(B), teacher_id(A), teacher_id(B), entry_date(A), entry_date(B), publish_date, semester and result_state.

3.4 Operating Environment

The website will be operate in any Operating Environment - Mac, Windows, Linux etc.

3.5 Design

Student activities have 3 steps -

- From Fill Up Process
- Courses Payment
- Student Profile

Top selected Student first fill his/her form, bank payment. After verification, student pays for their selected courses. Then he can enter his profile.
Every student profile contains his/her personal information, results, taken courses, dropped courses and notice.
Notice will contain all the news of IICT.

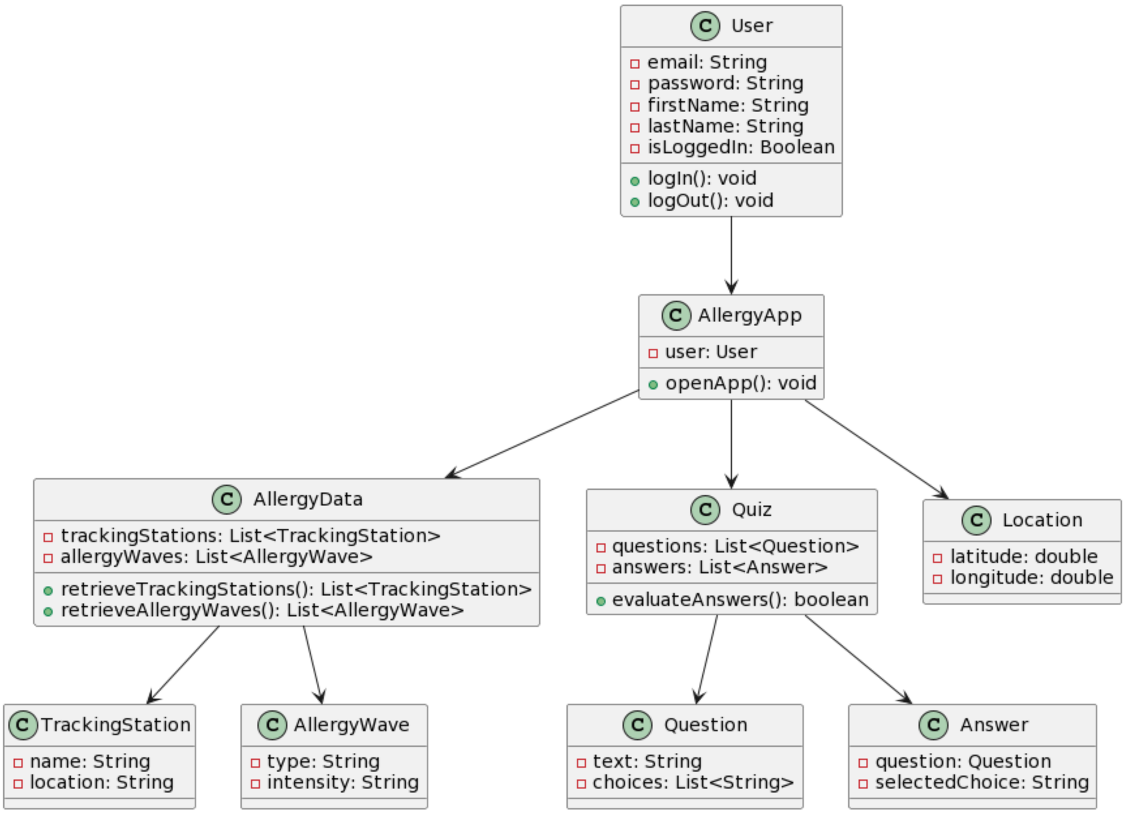

Figure 3.2: Student Activities

Teacher activities have 2 steps -

- Director
- Course Teacher

Director can re-view the result, publish result, give notice and also create teacher. He can also perform course teacher activities.
Teacher creates results, view students and create notice.

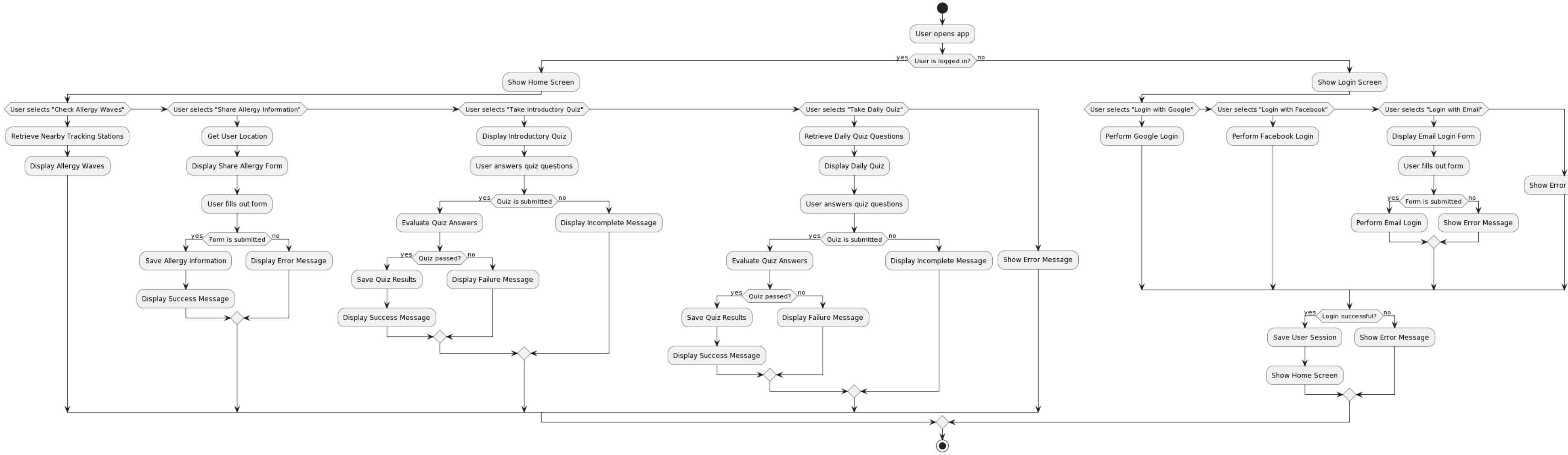

Figure 3.3: Activity diagram

4 Requirements

”IICT WEBSITE” is a result processing web software. So the main art of this product is to enter data of results and publish.

4.1 Business Requirements

should be from the point of view of the sponsor, business objectives The software shall provide user management functionality, allowing administrators to create and manage user accounts with different roles and permissions. Additionally, we

4.2 User Requirements

4.3 System Requirements

Table of contents for the System Requirements:

Non-functional requirements			**Functional requirements**	
ID	**Type**	**Name**	**ID**	**Name**
NFR-1	Performance	Memory size	*FR-1*	Health metrics alert
NFR-2	Performance	Server (processor) speed	*FR-2*	Forecast alert
NFR-3	Performance	Regularity of database updates	*FR-2*	User alert
NFR-4	Design [constrains]	Platforms	*FR-3*	Registration
NFR-5	Design [objectives]	Maintenance	*FR-4*	Welcome quiz
NFR-6	Design [decisions]	OS-specific	*FR-5*	Daily quiz
NFR-7	Design [decisions]	Databases	*FR-6*	Show login page
NFR-8	Design [decisions]	API	*FR-7*	Show main page
NFR-9	Design [decisions]	Cryptography	*FR-8*	Show map
NFR-10	Security	Security, access	*FR-9*	Show articles
NFR-11	External Interface	Hardware, accessories	*FR-10*	Upload articles
NFR-12	External Interface	Hardware, databases	*FR-11*	Sort articles
NFR-13	External Interface	Software, authentication	*FR-12*	Request allergens data
NFR-14	External Interface	Software, paying services	*FR-13*	User permissions
NFR-15	External Interface	Software, health tracking apps	*FR-14*	Show FAQ
NFR-16	External Interface	Software, databases access	*FR-15*	Contacts
NFR-18	External Interface	Software, permissions		

4.3.1 Functional Requirements

WHAT SYSTEM SHOULD DO –¿ input + output + conditions –¿ specify expected behavior during potential unexpected inputs unambiguous (if personal information –¿ specify)

TODO: add unexpected behavior

Functional requirements		
ID	**Name**	**Description**
FR-1	Health metrics-based alert	If the system recognizes abnormal metrics values received from the connected apps accessories, the user receives a notification suggesting to submit an alert.
FR-2	Forecast alert	If nearby stations show high levels of allergens identified in the user profile for tomorrow, a user gets a notification warning to be ready for the possible allergic reaction.
FR-2	User alert	User should be able to create an alert by clicking on the corresponding button on the main page. To view the next steps user has to press the ”next” arrow, on the final step he should press ”Submit an alert” button. 1) Once the button is pressed, user should be redirected to the page with a zoomable map with a pin with the current location. User can change the location of the pin by holding it and dragging it to the new location. 2) On the next page he/she will be suggested to choose allergens of select ”Not sure” if he/she does not know exactly which to allergen he/she got the reaction. 3) Next page will contain symptoms that user can select from and a text box for user comments (<50 characters). User can then finish the procedure by clicking the ”Submit an alert” button and should be redirected to the Map section with zoom on the created alert.
FR-3	Registration of the new user	Once user logged in using either email or social network, he/she should be redirected to the user profile page, where he has to specify the following details: 1) username that will be shown to other users: alphanumeric string from 4 to 8 characters 2) password to login using an email: alphanumeric string from 4 to 10 characters 3) date of birth 4) email should be automatically added if a person logged in using an email, if not it should be retrieved from the social network details 5) profile picture: user can choose a profile picture from some suggested options
FR-4	Welcome quiz	Once a new account has been registered, a user should fill in the Welcome quiz with questions on his allergens, general condition, and how often he/she has severe allergic reactions and other questions.
FR-5	Daily quiz	If a user enters the app first time for the day, he/she gets a pop-up window with daily ”How are you feeling?” test. If he/she chooses that he/she feels bad, the system suggests creating an alert.
FR-6	Show login page	Once a user opens the app, the login page should appear. It should look like the orange background with a logo and write text boxes. In the middle, there will be a box for the nickname/email/buttons of Google and Facebook. Lower, there should be a box for password. Below these boxes, there should be links ”Forgot the password”, ”Sign up as a new user”. 1. If person writes correct information in both fields or logs in using social network (into already existing account), he/she should enter the main page. If either of fields fail, system erases info from text boxes and shows warning. 2. If person fails to enter correct credentials for 3 times, new warning should appear suggesting to click ”Forgot the password”. 3. If person clicks ”Forgot the password”, he should be redirected to the page with only text box for email. 3.1 If entered email exist, system should send the link for updating password to the entered email/ 3.2 If the email does not exist, system gives a warning with suggestion to register as a new user. 4. If user clicks on ”Sign up as a new user”, system redirects he/she to the register page, where he has to input information as indicated in FR-3.

FR-7	Show main page	On the main page user shall see the following information: 1) today's and tomorrow's weather (temperature, humidity, wind, rain/no rain) - shall be retrieved from the Google Weather 2) intensity of the allergens, specified in his/her profile, for today and tomorrow from the closest station to user location 3) featured article of the day 4) Button to submit a user alert (FR-2) 5) link to the profile section 6) link to FAQ
FR-8	Show map	User should be able to view the map section which will contain the interactive map with two layers, that can be activated or deactivated. 1. The first layer, stations, shall contain the information from the allergy stations about yesterday, today, and tomorrow's prognosis. Information from stations will look like a pinpoint on the map with a color corresponding to the intensity of the signal from the station (from green to red). User can click on the pinpoint to view full information of the alert containing: a) intensity of allergens on the 10-scale for each allergen specified in his profile for yesterday, today, and the prognosis for tomorrow b) special comments from the station if applicable 2. The second layer should contain the other users' alerts for the allergens specified in the user profile. They will look like exclamation marks on the map and will contain the following information: a) name of allergen/allergens b) date and time of the alert c) nickname of the user who submitted it d) user's comments on symptoms The map should be centered according to the user's location.
FR-9	Show articles	Users should be able to view the articles section, will all corresponding materials and pictures. The articles section will contain the Name of the article, Keywords, and a one-sentence description. The section should be organized as a list to scroll, containing hyperlinks to articles' pages. Once a user opens the article page, the return button should appear at the top.
FR-10	Upload articles	Experts should be able to upload their articles via the special form, so it will be sent to the redactors
FR-11	Sort articles	User should be able to sort articles according to the date published, topic or keyword
FR-12	Request allergens data	App should request new data from the database an update the table with user and station alerts on the daily basis
FR-13	User permissions	After first log in to the app user shall receive the notification with suggestion to give the app permission to: 1) access the smartphone location services 2) connect to installed health-tracking apps 3) get notifications from the app
FR-14	Show FAQ	User should be able to view the Frequently Asked Questions section of the main menu with the toggle list of main questions on data sources, app functionality and user policy
FR-15	Contacts	User should be able to contact developers with suggestions, corrections and comments on the app interface and content

4.3.2 Nonfunctional Requirements

HOW SYSTEM SHOULD DO IT

4.4 Performance Requirements

4.4.1 Static Requirements

- memory size - web server should work with 100000 simultaneous users

4.4.2 Dynamic Requirements

"IICT WEBSITE" will be used for result system of IICT programs, like - PGD, MIT. So for more interaction .NET, React and MongoDB is used.

The web server should provide a response in less than 3 seconds

when th submit button is pressed the workspace should load within 2 seconds

4.4.3 Design Constraints

* we have to select the group of platforms: Android and IOS from version X *

4.4.4 Design Objectives

* user friendly * we want to maintain the app for 5 years at least

4.4.5 Design Decisions

The "IICT WEBSITE" website is being build on .NET framework, C# language, React, React Redux, JavaScript and MongoDB.
Back-End - .NET framework, C# language.
Font-End - React, React Redux, JavaScript.
Database - MongoDB.

- **OS-specific**
  We will have two versions of our app for two platforms: Android, written in Kotlin and/or Java; and IOS, written in Swift.

- **Databases**
  We will have all our databases stored in the Oracle system, accessible via SQL calls with provided entry requisites (login, password)

- **Specific API**
  We will use API from National Allergy Bureau (AAAAI) to retrieve the data for the regions of North and South America, and EAN provided resources to retrieve the data for the European region. We will store this data in the Oracle database, for more detailed information on the database structure, please, refer to 3.3.

We will retrieve information from the Database, Alerts, and Stations tables on a daily basis to display in the App interface. We will also update information from the Alerts table each hour, so users will be updated.

- **Cryptography**
  Encryption is the process of encoding all user data on a device using symmetric encryption keys. Once a device is encrypted, all user-created data is automatically encrypted before committing it to disk and all reads automatically decrypt data before returning it to the calling process.

- **Algorithms**

4.5 Security Requirements

In order to access the app and perform all possible actions users have to be registered in the app. We will also use an API to access the public or private databases. First, we will maintain the Oracle database with data from monitoring stations which will have two modes of access, both only for users-developers with special login and password data. The first group of users will be developers-editors, who will be allowed to edit the database (by some software), and developer-retriever with only view-retrieve permissions. All users' data will be stored in the database which only developers can access.

4.6 Software Quality Attributes

In the development phase also testing and conferences of users is been continued. So that the quality of the software is been maintained and all the requirements are been fulfilled.
Database, logical and also UI test is required.

4.7 External Interface Requirements

There are several requirements related to the external interfaces such as hardware and software. The app will interact with several external databases to infer the set of map points, predictions, and alerts from the user. There are also some health-related apps we want "Breathe Easy" to interact with. Apart from software, we want our app to interact with SmartWatch (Apple Watch, Samsung Watch, etc.)

1. Hardware:
   - access to SmartWatch
2. Software:

- authentication system from Google and Facebook
- authentication using email service (ability to send an email with sign-in access link)
- connecting to PayPal and other banking services to purchase the full version of the app
- access to health tracking apps (such as Apple Health, Samsung Health)
- connection to the database of users (Oracle)
- connection to the database of monitoring stations (Oracle)
- phone notification permission, location