Authors

Johan Andersson
David Fogelberg
Sam Halali
Gunnar Gunnarsson
Nandha Gopal Elangovan
Miguel Angel Sanchez Cifo
Jonathan Granström

Git repository: https://github.com/lol2kpe/EDA397_Team3
Issue tracker: https://github.com/lol2kpe/EDA397_Team3/projects/2

Project Description

Project Title

H4U

Platform

Android

Minimum and target SDK

5.0

Description

The H4U app helps a user search for hospitals, pharmacies and doctors including; doctor details, doctor availability and doctor reputation. It will also show the directions to each hospital by showing the route map. The route map will be set from a user's current location and will show both a visual and descriptive way of the route, which will make it easy for a user to find the hospital location.

Sprint Log: Sprint 2

Commitment

List the features/stories the team commits to finish during the sprint.

User Stories

ID	User story
47	As a user, I would like to find directions to a hospital, so that I know how to get to a selected hospital
46	As a user, I want to see information about shown hospitals, so that I know more about a selected hospital
45	As a user, I want to search or filter by injury or problem, so that I can find relevant hospitals
44	As a user, I want a login activity, so that I can create a profile or validate my credentials
43	As a user, I want to see and change my search radius, so that I can see hospitals in my area

Enhancement

ID	Enhancement
49	As a developer, I want to query the database by filter options, so that I can find the relevant objects to show the user
50	As a developer, I want a better class structure, so that it is easier to filter and search for relevant objects

Tasks

ID	Tasks
48	Create a "populate map"-function
51	User profile creation and login
54	Add more services to service class
55	Fix database fetch data
56	Create dummy data generator
57	Change position of filter button to show directions
58	Update database fetch for Place
59	Create save for user profile
60	Implement SQLite for save
61	Create better visualiztion of selected objects on the map
62	Add "get directions" functionality to selected map object
63	Add "problem/injury" filter option
64	Add proper search functionality
65	Add functionality to user profile
66	Add "medicine reminders" functionality
67	Add "settings" functionality
68	Add possibility to search by radius (or to set the radius)
74	Automatic zoom in to position in map

Work Done

Feature	Commits	Group members	Effort	Practices
Create a “populate map”-function	22b5de5	Gunnar GunnarssonSam Halali	3hr	Refactoring, Pair programming
User profile creation and login	202d397 56f3a79 bf4241e 1665fe4 0381c82 caef8ed	David Fogelberg Miguel Angel Sanchez Cifo Nandha Gopal Elangovan	14hr	Test-first .Refactoring, Pair programming
Refactor data-model objects	4c2427b 6c51291 12e8a07	Sam Halali Johan Andersson	5hr	Refactoring, Simple-design

Feature	Commits	Group members	Effort	Practices
Create a representation of opening hours	155bc41	Sam Halali	5hr	Simple-design ,Test-first
Create dummy data generator	b20a1de	Sam Halali	2hr	Refactoring, Simple-design
Create better visualization of selected objects on the map	0d51318	Sam Halali Miguel Angel Sanchez Cifo	4hr	Pair programming
Add “get directions” functionality to selected map object	0d51318	Sam Halali Miguel Angel Sanchez Cifo Johan Andersson	4hr	pair programming, refactoring, simple-design
Add possibility to search by radius (or to set the radius)	e12702e	Nandha Gopal Elangovan	2hr	test-first
Refactor FilterActivity	0155388 224fb4c	Jonathan Granström	12hr	Refactoring, code standard, sustainable pace, continuous integration
Add Tabs to FilterActivity	bdd672c	Jonathan Granström	6hr	Refactoring, code standard, sustainable pace, continuous integration
Create save for user profile	c6a3d09	David Fogelberg	3hr	Continuous integration
Implement SQLite for save	c6a3d09	David Fogelberg	3hr	Continuous integration
Update database fetch for Place	c6a3d09	David Fogelberg	1hr	Refactoring
Add possibility to filter by symptoms/injury	9da9d0d	Jonathan Granström	2hr	Simple design
Automatic zoom in to position in map	caef8ed	Miguel Angel Sanchez Cifo	1hr	
Add Injury typ filter option	d68b62b	Johan Andersson	2hr	code standard

Reflections

In this section we describe our experiences of implementing agile methods in our project during sprint 2.

The most important factor was “Managing the Customer and her expectations”, which was also the most challenging one. After the acceptance test for the first sprint, we carried out an analysis session based on the customer’s feedback. We figured out the actual improvements needed for the next acceptance test. In regard to implementation of agile principles, we had some suggestions from Magnus.

After the analysis, we figured out the main problems in sprint one were

1. Identifying dependencies in the functionality.
2. Communication in the group.
3. Sprint planning.

The sprint started off with a sprint planning meeting, where we discussed and decided about the target backlog. Also we clarified the sprint requirements and proceeded with the user stories and tasks. The major decisions made in first meeting were

1. Create user stories based on customers feedback, and assign priority to them.
2. Identify the dependencies between the user stories and tasks.

After the easter break, we had a sprint meeting where we created the task list and delegated the tasks amongst us.

Implementation of agile principles:

In order to manage requirements and organize the activities, we had to make the most out of the time and resources available. We focused on implementing agile methodologies.

The methodology

1. Iterative and incremental process
2. Test-first: We started off with test-first development, but it was really hard to do correctly. This mainly comes from the fact that programming in general meant a lot of experimentation and uncertainty before reaching the optimal solution. Creating tests was difficult, as it was hard to predict if they would be valid and properly represent the actual code down the road. The test-first principal did not work well when implementing the save function for the user profile. The reason for this is it was rather difficult to test the local SQLite database without actually inserting real values. When the values had been inserted it was easy to just log everything in the local database and see if everything was saved correctly.
3. Coding standards: Coding standards were decided at the initial stage of the project. Which was easy to implement by using "Ctrl + Alt + L".
4. Pair programming: It is an intense experience, and it requires a bit of time to get used to. Where some pair programming was done, we had to connect the work with other members' work. However, it worked well when creating a better visualization of selected objects on the map.
5. Sustainable pace: It worked very well when it came to populating the map with hospitals. However, there are some user stories which were not achieved during this sprint and this could be for having and slow sustainable pace but also miss plan since some user stories took longer than expected.
6. Code Refactoring: In order to keep the code clean and flexible a lot of refactoring was done.
7. Collective code ownership: Every member of the team had the access to change anything in the code to fulfill new requirements, or refactoring odd-looking code. It helped also in enforcing the coding standards.
8. Simple Design: We have used this method since we had ongoing activities, which may involve refactoring and design decisions. This mitigates the common risk of overdesign. It worked well with certain design patterns.
9. Continuous-integration: Since we worked separately on same code portion for several times we had time to debug at the end of long integration cycles, so we have used the Continuous-integration principle to produce a clean build of the system for several times.
10. Testing: In order to verify that the system worked as expected and without breaking the functionality of the code, we carried out extensive testing for each functions.

These were agile principles considered in this sprint, apart from these principles, we had some complementary methods for achieving the project. Our project consists of technical tasks which has design, coding, and testing.We had appropriate methodologies to apply for each task.As the project keep on growing we were able to increase the development speed and improve the functionality. We had sprint meetings which were organized in a scrum fashion to achieve consistency in our team work. In few portions we used available code and libraries instead of writing our own so that we were able to time need to develop.we had few drawbacks due to easter holidays during this sprint, so we had time constraint, there were few tasks been postponed to next spring but we have achieved the prioritised tasks and user stories. With our experience, we believe that "The agile methodologies is to make software development faster and, at the same time, have higher code quality, higher customer satisfaction". and also we understood that At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.Thus these are the reflections made from our work in sprint two.