Contents

1 1.A. Hand-to-Hand Drone delivery

TA Assigned: Halima Najibi[1], Marina Zapater

Objective: Develop an app that programs a drone to deliver items to a receiver based on his/her geolocation

Material required: Tablet/Phone, Smartwatch, Drone

Minimum requirements:

- Firebase or Google authentication - *ONLY REGISTERED VALID USERS CAN CONTROL THE DRONE AND SEND ITEMS TO EACH OTHERS*
- The sender can select a receiver and see his/her location on the map (GPS coordinates from the Smartwach), then he can send an item to him/her
- Send notification to the receiver with the delivery request (providing some details about the delivery: object to be delivered, estimated time of arrival, etc.).
- The receiver can *ACCEPT* or *REFUSE* the delivery
- Once a delivery request is sent and accepted by the receiver, the App calculates the trajectory to be taken by the drone, as well as the estimated time of arrival (ETA)
- The sender (tablet) can see a *LIVE* status of the drone's and receiver's positions, and the ETA
- The sender and/or the receiver can *CANCEL* the delivery at any time, in this case the drone should return to the sender (GPS coordinates from the tablet)
- In case the connection with the receiver is lost (Smartwatch disconnected/off), the delivery is automatically canceled and the drone should return to the sender

Optional Features:

- *LIVE* update of the trajectory of the drone and ETA following the movement of the receiver
- Keeping a history of past deliveries (completed, cancelled, aborted), using Firebase. Possibility to see/clear the history from the tablet.

[1] <halima.najibi@epfl.ch>

2 1.B. Drone Photography

TA Assigned: Halima Najibi[2], Marina Zapater

Objective: Develop an app to take pictures/videos using a drone

Material required: Tablet/Phone, Drone (optionally the Smartwatch)

Minimum requirements:

- See the output of the drone's camera in REAL-TIME in your tablet/phone
- Control the drone from your tablet and move it to get your perfect view
- Use the tablet/phone to capture the photos
- Take pictures and record videos from the tablet
- Zoom in and out using touch screen in the tablet/phone. Since the camera lens of the drone itself is fixed focus, You can implement the zooming physically by by moving the drone closer or further from the ground or object. Alternatively, you can zoom in electronically by cropping the picture and interpolating.

Optional Features:

- Quick shot using the Smartwatch (button or specific hand gesture).
- Automatically save the shots in a cloud storage (Firebase).

[2] <halima.najibi@epfl.ch>

3 1.C. Gesture-Controlled Drone

TA Assigned: Halima Najibi[3], Marina Zapater

Objective: Control a drone using predefined hand gestures detected by the Smartwatch

Material required: Smartwatch, Tablet, Drone

Minimum requirements:

- Control the drone using *SPECIFIC HAND GESTURES* to:
  - fly
  - land
  - move up/down
  - move left/right, etc..
- The gestures are captured using the *motion sensors* of the Smartwatch
- Two options: 1 - the watch directly sends commands to the drone 2 - it is connected to the tablet via Bluetooth and continuously sends its location to the tablet/drone (when fly mode), then the tablet commands the drone.
- Drone does not respond to abrupt gestures (for example if the person falls or is attacked by a bee :D)
- *SAFETY LANDING* in the following cases:
  - Lost connection to the Smartwatch
  - Battery of the drone is low
  - Inactivity of the user (Smartwatch)
  - Too many abrupt gestures from the user

[3] <halima.najibi@epfl.ch>

4 1.D. Train with your Drone!

TA Assigned: Halima Najibi[4], Marina Zapater

Objective: Make the drone your training assistant

Material required: Smartwatch, Drone (optionally Tablet)

Minimum requirements:

- The drone follows you while you are doing you training -on land- such as running, climbing, etc...
- When you start your training, the drone continuously receives your location and moves following it
- The drone can be carrying some of your personal items (keys, wallet, water etc...), so you don't have to carry them
- You can command the drone to land, so you can take some water, an item, etc..
- The drone can take pictures/videos of you (when climbing for example)
- The pictures taken by the drone are automatically saved and sent to a cloud storage (Firebase)

Optional Features:

- All the commands to the drone can be done using hand gestures, which are detected using the motion sensors of the Smartwatch
- Connect a tablet/phone to see the LIVE view from the drone when in motion
- In this case you can send the drone ahead of you to inspect a path when you are not sure if it is safe or not, so you can decide on which way to proceed

[4] <halima.najibi@epfl.ch>

5 2.A. Wellness Monitoring Data Processing

TA Assigned: Elisabetta De Giovanni[5], Marina Zapater.

Objective: Processing past fatigue/activity data and propose new recordings

Useful links: https://esl.epfl.ch/WellnessMonitoring

Material required: Tablet/Phone, Watch, HRM

Minimum requirements:

- Firebase authentication
  - Ids of already created Firebase
  - The user must be already present in the Firebase, otherwise error
  - User Login or Admin Login
- History admin
  - Read history data from Firebase (the DB will be given to the students, anonymous or example data will be generated). History data includes:
  - RR interval (i.e. heart rate (HR)) collected with an HR monitor during a fatigue test;
  - HR and exercise duration measured with a smartwatch.
- History user (similar to app we have, integration?)
  - Read history data from Firebase with specific login
  - Same data as admin
- Data visualization admin
  - Apply algorithm for fatigue detection (given to the student)
  - Compute TRIMP score for activity (equations given to the students)
  - Group subjects for fatigue/no fatigue, HR during activity, exercise duration and TRIMP score
  - Show bar graphs, trend plots, etc.
  - MUST be user-friendly, interactive and nice to look
- Data visualization user
  - Show fatigue detection

[5] <elisabetta.degiovanni@epfl.ch>

  - Show TRIMP score for activity
  - Show bar graphs, trend plots, etc. for fatigue/no fatigue, HR during activity, exercise duration and TRIMP score for specific user
  - MUST be user-friendly, interactive and nice to look
- Activity/fatigue correlation admin
  - Show in another plot an activity/fatigue correlation positive or negative (the algorithm will be given to the students) for different subjects and within the groups.
  - If a correlation shows up, propose a new fatigue test to specific user/users
  - Propose new activity trackings
- New recordings user
  - Accept/reject proposal of new recording (Polar monitor BLE) to check for problems or benefits (the recording will be saved on the same Firebase where the history is).
  - Accept/reject new activity tracking (watch sensors).
  - If the person is outside, the activity recording must be stored on the watch and uploaded on Firebase once there is a connection.

Additional features:

- Integration with existing recording app

6 2.B. Team Sport Match Monitoring

TA Assigned: Elisabetta De Giovanni[6], Marina Zapater.

Objective: Monitoring a team sport match in real-time

Material required: Tablet/Phone (for 'coach'), Watch1 (player 1), Watch2 (player2)

Minimum requirements:

- Tablet/coach login
- Firebase Authentication
- Home tab/page with history of previous matches
- Strategy tab/page
  - Field map layout
  - Possibility to trace a path to make during the match
  - Notes to insert strategy of take overs or other events at specific time
- Recording of new match tab/page
  - Button to connect watches (players) that opens recording activity on them
  - Field map layout
  - Players markers tracking position on the map and vital signs (HR) shown on top
  - Field to add current score
  - Past events shown on the right side to keep track of the live strategy
  - Strategy previously prepared on the left side
  - Possibility to swipe strategy events from left to right (event happening)
  - Sending short messages or alarms to the players for events (like 'next set, player 1 take over')
  - When match ends, save it in history and show events list and some stats (avg, min, max) on the players HR
  - Save matches and stats in Firebase (if there is no connection, save locally)
- Watch live match recording
  - Track HR and GPS position to send it to the tablet
  - Buttons to send messages to the coach (ex. green: everything is ok, orange:

[6] <elisabetta.degiovanni@epfl.ch>

need of take over during the next set, red: request time out)

    - Receiving messages from the coach/tablet (short notifications and alarms, ex. vibrations code established in strategy)
    - Save HR, GPS, duration of match in local if connection is lost between tablet and phone and send info when connection is re-established
    - Stop recording if matches finish or after take over (can be a button)

- Layouts MUST be user-friendly

Additional features:

- Possibility of choosing different sports, strategies and maps
- Prebuild strategies to download from Firebase and load for a specific new match
- Automatic connection of watches of the players who take over

7 2.C. Personalized long-term athlete training

TA Assigned: Elisabetta De Giovanni[7], Marina Zapater.

Objective: Helping athlete in organizing their training based on physiological signals

Useful links https://www.hexoskin.com/pages/developers

Material required: Tablet/Phone, Watch, Smart t-shirt ECG (HRM, if not possible)

Minimum requirements:

- Tablet
  - Firebase or Google authentication
  - History of recordings: show type of trainings per week, some stats (avg, max, min) HR, duration of trainings, speed.
  - If there is a stress test (see below), button to compute ventilatory thresholds (VT1, VT2) and VO2 max (algorithm given to the students), in BPM
  - Show suggestions for next training based on previous trainings (if endurance was done one day, suggest a short sprint recording, setup duration arbitrarily).
  - Button to accept suggestion and send the type of training and duration on the watch. If rejected let the subject choose the type of training on the watch
- Stress test on watch
  - Connect smart t-shirt to watch
  - Two types of recordings:
    1. 30 minutes increasing stress test: intervals of 3 minutes increasing speed running or power of bicycle. For testing only 30 seconds is fine.
       - ∗ Collect RR intervals from smart t-shirt ECG and HR from the watch
       - ∗ Collect GPS for speed computing
       - ∗ Show intervals on the watch and alarms to increase the speed.
       - ∗ Show speed on the watch for the subject to check.
       - ∗ Show the time of the recording
       - ∗ Save data locally if connection between tablet and watch is not established

[7] <elisabetta.degiovanni@epfl.ch>

- ∗ When there is connection, send RR intervals, HR, speed and time

2. Training session:
   - ∗ Choose three levels of training: 1) endurance, 2) sprint, 3) High-intensity interval training (HIIT)
   - ∗ Start recording collecting RR, HR, GPS (for speed), and record/show time
   - ∗ Different set ups for the different trainings: 1) RR (or HR) should be in the range of VT1, 2) RR (or HR) should be in the range of VT2, 3) alternating short intervals at range of VT1 and range of VT2
   - ∗ Save data locally if connection between tablet and watch is not established
   - ∗ When there is connection, send RR intervals, HR, speed and time

Additional features:

- Possibility to change smoothly from day to week to month (to more) history plotting

8 2.D. Tracking Application Pro for runners

TA Assigned: Dionisije Sopic[8], Marina Zapater

Objective: Running application with audio coaching

Material required: Tablet/Phone, Watch/Polar belt

Minimum requirements:

- User login page (Sign in/Sign up)
- Firebase authentication
- Making running groups
  - Making comparisons between team players for each kilometer and in total
- Running history
  - Number of kilometres
  - Running duration
  - Number of burnt calories
  - Average time
  - Showing on the map the exact path with different colors (each color corresponds to the HR value in a certain interval of time)
  - Plotting the value of the HR
- Vocal coach
  - After each kilometer, the vocal coach informs us of the distance
- Watch:
  - After each kilometer, the distance and the average time are sent to the watch

[8] <dionisije.sopic@epfl.ch>

9 2.E. Training with AR

TA Assigned: Elisabetta De Giovanni[9], Marina Zapater.

Objective: Real-time sport training with AR

Useful links https://tech.moverio.epson.com/en/bt-300/

Material required: Tablet/Phone, Watch, Moverio AR glasses

Minimum requirements:

- Firebase authentication
- Real-time training
  - Set goals on the tablet before training and send them to the glasses
  - Collect data from watch (HR, GPS, compute speed, time, ...)
  - Show data in real-time to the glasses (current recording)
  - Show score previous training (to beat and create records by changing speed)
  - Save data in local (SQL) if there is no connection
  - Send data to Firebase as soon as connection is established
- Glasses sensors activity tracking
  - Collect accelerometer data from glasses and detect crashes (as discussed with the students)
- History trainings
  - Read data of the previous training from Firebase and show in the tablet. History data includes all the data shown in the glasses
  - Show HR/activity correlations (algorithm given to the students)

Additional features:

- Take pictures from the glasses during activity and share it with friends

[9] <elisabetta.degiovanni@epfl.ch>

10 2.F Sports — Ski Training with AR

TA Assigned: Elisabetta De Giovanni[10], Marina Zapater.

Objective: Real-time sport training with AR

Useful links https://tech.moverio.epson.com/en/bt-300/

Material required: Tablet/Phone, Watch, Moverio AR glasses

Minimum requirements:

- Firebase authentication
- Position tracking (GPS).
- Real time display:
  - Speed
  - Altitude
  - Heart rate
- App functionalities:
  - Select the slope/destination and then a navigation assistant guides the skier towards the destination.
  - Top speed recording for a selectable time interval (interesting for training).
  - Save data to server or in local if no connection.
  - Emergency calls (automated option: "Are you ok?!" -> if no answer, emergency call thrown).
  - Take pictures.
  - Share/visualize localisation of other users of the app.
  - Call people directly through the glasses interface (no need to use another device).
  - Meteo infos.
  - Show ski domain information (occupancy, open/closed, slopes, infrastructurs, ...)
  - Dangers (avalanches, ...)

[10] <elisabetta.degiovanni@epfl.ch>

11 3.A. ECG (or general signal) Annotator

TA Assigned: Elisabetta De Giovanni[11], Marina Zapater.

Objective: Recording ECG, finding its main points and detecting the quality of annotations (for doctors and patients)

Useful links: http://cvrgrid.org/tools/waveform-ecg

Material required: Tablet/Phone, Watch, Inyu or BLE ECG device

Minimum requirements:

- Patient login
  - User registration with patients right
  - Interfacing with a bluetooth ECG device
  - ECG recording and plotting
  - Sending ECG to a specific doctor in the list of users (External Server read/write)
  - If no connection save ECG locally and send when the connection is established
  - Show notifications from doctor in case of problems
- Doctor login
  - User registration with doctor rights
  - Notifications of new recordings
  - Accessing ECG recordings of his/her patients only (External Server)
  - Ask if the recording should be saved locally in case of future missing connection
  - Choosing manual or automatic detection of points
  - Manual annotation: tap points on signal and insert small description; insert possible anomalies; editing with a popup question ”are you sure?”
  - Automatic annotation: button to run the delineation (algorithm given to the students), possibility to edit with a popup question ”are you sure?”
  - Button to check quality of annotation (algorithm given to the students) showing for each annotation (manual/automatic) the percentage of ”sanity”
  - Propose to share annotations with other doctors in anonymous if the sanity < 50
  - Send notification to patient in case of problems

[11] <elisabetta.degiovanni@epfl.ch>

- The code MUST be parametric (easily editable)
- The interface MUST be user-friendly

Additional features:

- Patient login:
  - HR recording from smartwatch and sending to doctor
- Doctor login
  - check HR recording and annotate anomalies

12 3.B. Real-time patient monitoring

TA Assigned: Dionisije Sopic[12], Marina Zapater

Objective: Real-time monitoring of patients using an online platform

Material required: Tablet/Phone, Watch

Minimum requirements:

- User:
  - User login page (Sign in/Sign up)
  - Firebase authentication
  - Chat with the doctor
  - Doctor's feedback should be shown on the watch
  - History of recordings: show the last doctor's advice, HR (avg, max, min) for a certain interval of time HR
  - Plotting selected features for a certain interval of time
  - Real-time plot (HR)
- Doctor:
  - List of all patients
  - History of advices, HR (avg, max, min) for a certain interval of time HR
  - Chat with the patient

Additional features:

- Loading a video from the firebase (This video can contain different exercises) and showing it on the tablet.

[12] <dionisije.sopic@epfl.ch>

13 4.A. Clone Dance!

TA Assigned: Grégoire Surrel[13], Marina Zapater.

Objective: Develop a rhythm game

Material required: Tablet/Phone, Watch

Minimum requirements:

The goal is to make an Android clone of the game *Just Dance!* (a rhythm game which requires the player to move in specific ways). We are limited in the ways we can get the player's position, so one possibility is to use a smartwatch on *each* wrist.

The signal processing could be not trivial for having a nice gameplay: it needs to be tolerant enough that the game is enjoyable and specific enough to be able to discriminate movements/positions.

A (extremely simple) in-game level editor is required because it will be useful when creating the project, and open the way to user-generated content.

Additional features:

- Sharing custom dances (song+level) online with Firebase storage
- Online hall-of-fame

Note

It is advised to initially start with an simple game design to develop the game mechanics and internals: an image of the next move and a bar going to zero when the move should be done. No need to start with a 3D world with an avatar of the player reproducing the current position with online capabilities if the data coming from the watch is unusable. Of course, it's desirable to improve the design with a 2D canvas or 3D context when the core is tested (and maybe even the editor working).

[13] <gregoire.surrel@epfl.ch>

14 4.B. Air drums set

TA Assigned: Grégoire Surrel[14], Marina Zapater.

Objective: Drums simulator

Material required: Tablet/Phone, Watch

Minimum requirements:

The idea is to have an air-drums set: with a smartwatch on each arm, simulating playing the drums in the air will make the tablet play sounds accordingly.

A set with 3 drums (left, center, right) is a good start.

Additional features:

- Have a five drums in the set

```
           drum 4         drum 5

   drum 1         drum 2         drum 3
```

Note

If the latency for playing the sound or feedback is too slow, consider turning it into a rhythm game: the sounds can be played a bit in advance and canceled if the player is way too late

[14] <gregoire.surrel@epfl.ch>

15 4.C. Sensiwall sports trainer

TA Assigned: Grégoire Surrel[15], Marina Zapater.

Objective: Design an app sor Sensiwalls

Material required: Tablet/Phone, Watch, Sensiwall screen

Minimum requirements:

The Sensiwall screens[16] are heavy-duty touch sensitive screens. The main goal of the project is to develop an app using this device, with great freedom about the exact use case. One possibility can be to reproduce the *Simon* game[17]: few tiles of color and the player has to reproduce a sequence which is each time longer.

Because of the multi-screen setup, integration of all inputs can be used to have a more complex game:

- 4 tiles on the Sensiwall
- 4 tiles on the tablet
- 2 tiles on the watch

Pushing the game further, the tiles could be within a screen, or even fly to another screen! At this point, colors would not be enough to distinguish the tiles, but adding a symbol (such as a family of *Space Inveders*) can solve the issue.

Additional features:

Consider a multiplayer game: both players get the same sequence (and tiles shuffling), the first one to complete it scores five points, the opponent has two seconds to finish to score two points. If a player makes a mistake, he can *not* finish the current round. The first one to reach 20 points wins the match.

[15] <gregoire.surrel@epfl.ch>

[16] <https://www.facebook.com/sensiwall/videos/797130410481555/>

[17] <https://en.wikipedia.org/wiki/Simon_(game)>

16 5. Ski Project

TA Assigned: Halima Najibi[18], Marina Zapater

Objective: Develop an app for ski and snowboard schools.

Material required: Tablet/Phone, Watch

Minimum requirements:

- The app must differentiate two type of users, namely, learner, and teacher.
- The learner:
  - has access to his own profile stored on a cloud
  - can modify his personal data (address, phone number, etc...), but not the level/skills. The level can be modified by a teacher only
  - has access to the scheduling (read only) to keep track of the lessons and to see when the next one is scheduled
  - can send an ”emergency” signal to teacher in case of accident, danger, etc...
- The teacher:
  - is a learner as well, and like a learner, he has is own profile with the right to modify his personal data
  - can notify his availability to give lessons, and he can see his scheduled activity
  - receives a notification (on both phone and watch) when a learner is assigned to him
  - can see the personal data of the assigned learners and have the right to update their level/skills
  - has access to a collection of exercises (read only) and can plan the activity with the assigned learner
  - can see a LIVE activity of the learner assigned to him (GPS location, HR..)
- The activity of both teachers and learners is tracked using the GPS of their devices and saved on a cloud

Additional features:

[18] <halima.najibi@epfl.ch>

- A third type of user may be considered, i.e., the manager, who:
  - is the one that schedules the lessons
  - has access to a database with all the available teachers of a specific school and the info of the learners
  - is able to assign the learners to the available teachers

17 6. Book library

TA Assigned: Grégoire Surrel[19], Marina Zapater.

Objective: Develop a library management app

Material required: Tablet/Phone, Watch

Minimum requirements:

There should be two apps:

1. Library app:
   - Scan barcodes of books to identify them (when adding new books and returning lent books)
   - Have an overview of the library with the book status
   - Have to capability to search for a book in the library
   - See the library users
2. User app:
   - Select the library the user is part of
   - In this library, search for books
   - When having a book, scanning the barcode can start the rental or end it
   - Get reminders/notifications pushed to the watch about the book borrowed (taken, returned, rental approaching expiry)

Additional features:

- Have a waiting list for books with notifications for the following user when a borrowed book comes back available
- The user B can take the book from user A without going back to the library for returning it. To be fair with the waiting list, the expiring date should not change, just the ”ownership”
- Handle items which are not necessarily books
- One might take items for an event (a course, a demo...)

[19] <gregoire.surrel@epfl.ch>

18 7. Travel Application

TA Assigned: Dionisije Sopic[20], Marina Zapater

Objective: Finding the closest meet up point for different users (The closest restaurant for instance)

Material required: Tablet/Phone, Watch

Minimum requirements:

- User login page (Sign in/Sign up)
- Firebase authentication
- Query for place information on a variety of categories, such as: establishments, prominent points of interest, geographic locations (Use the Places API)
- Based on the current location of different users (latitude and longitude), find the closest meet up point
- Show the path on the map (Between the user and the meet up place)
- Once the user presses on the button RECORD, we start getting the data from the watch (heart rate HR)
- Plotting the HR value in real-time
- The exact meet up place is sent to the watch as well as the average value of the HR during the walking phase
- Meet up history

Additional features: - Navigation (From the user position to the meet up point)

[20] <dionisije.sopic@epfl.ch>

19 8. Solar alarm

TA Assigned: Grégoire Surrel[21], Marina Zapater.

Objective: Create an alarm-clock which is sun-based

Material required: Tablet/Phone, Watch

Minimum requirements:

Provide all functionalities of a normal alarm-clock: display progression in the day (solar day rather than timezone-defined 24h day), set alarms to some specific events such as one saying ”*Go to sleep* 9h before sunrise”, ”*Lunch* at noon”.

This app is therefore location-dependent, and additional input should be provided because of the change of rhythm between winter and summer.

The *NOAA* spreadsheets[22] can be used to compute the sun’s position throughout the years.

Additional features:

- Offer the capability to handle multiple different locations on Earth.
- Have a very graphic user interface to *see* how alarms are set rather than plain-text rules such as sentences like ”*Something* 2h after noon”

[21] <gregoire.surrel@epfl.ch>

[22] <https://www.esrl.noaa.gov/gmd/grad/solcalc/calcdetails.html>

20 9.B. Robot rescure mission

TA Assigned: Grégoire Surrel[23], Marina Zapater.

Objective: Control a exploration robot for rescue missions

Material required: Tablet/Phone, Watch, TI-RSLK kit

Minimum requirements:

Develop an application that allows the robot to explore (automatically or remote-controlled) and return safely back to the operator (that is, to exit a certain building). For robustness, the watch can be used to take over a dead tablet.

The Bluetooth SDK is available online[24].

- Provide a manual control of the robot
- Have an automatic exploration mode. The smarter the better!
- Create a live map of where the robot went, and obstacles it found with its sensors
- Get the robot automatically out
- If the tablet fails (empty battery, dead hardware...), the watch must take over:
  - Controls with tilting the watch
  - Map in fullscreen
  - Tapping the map asks if the mission has to be finished and autopilots to the end

[23] <gregoire.surrel@epfl.ch>

[24] <http://users.ece.utexas.edu/~valvano/android/>