RDroid

מסמך עיצוב

שמות הכותבים:

יניב שרון

יובל ינושפולסקי (ראו הערה בעמוד 3)

גרסה: v0.1d

תאריך: 2/12/2015

היסטוריית גרסאות המסמך:

|  |  |  |
| --- | --- | --- |
| **תאריך** | **גרסה** | **תקציר שינויים** |
| 02/12/2015 | v0.1d | * המסמך נוצר. |
|  |  |  |

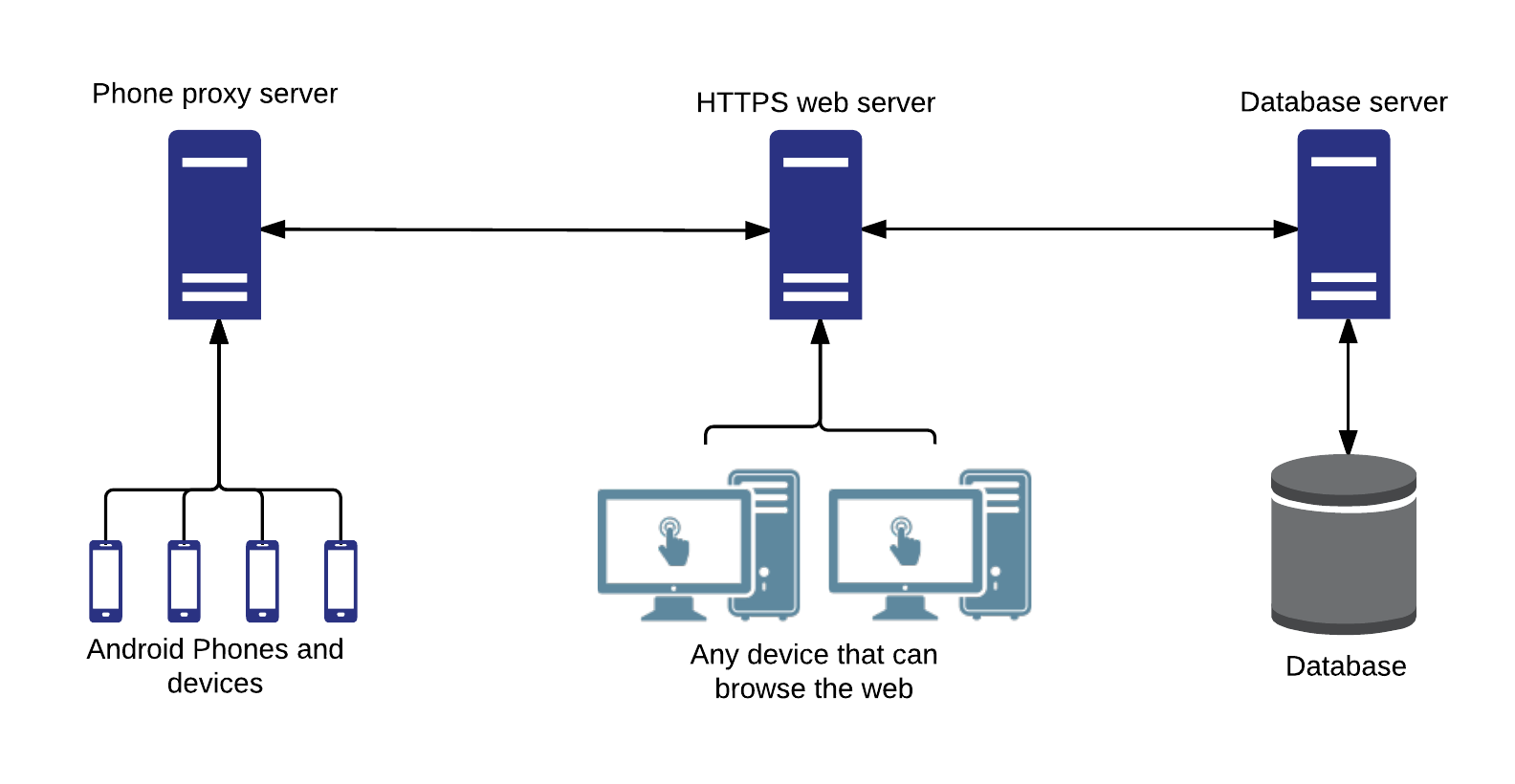


**הקדמה**

מטרת המסמך הזה היא תכנון מקדים של הפרוייקט RDroid. במסמך זה יתוארו המודולים, הפרוטוקולים והמבנה הכללי של המערכת.

קהל היעד של הפרוייקט הינו כל אחד שברשותו מכשיר Android וברצונו לנהל אותו מרחוק בעזרת המחשב הנייח שלו (או כל מכשיר אחר).

RDroid הוא שירות שייתן ללקוח לנהל את הטלפון מהמחשב. הלקוח יתקין את אפליקציית השירות בטלפון ויתחבר למשתמש הפרטי שלו בעזרת שם וסיסמא. כאשר הלקוח ירצה לנהל את המכשיר, הוא יתחבר לאותו המשתמש דרך האתר ויוכל לנהל את המכשיר משם.



הפעולות הניהוליות שינתנו מהשירות הן:

1. קונטיינר מוצפן: השרת יצפין קבצים על הטלפון ותהיה גישה אילהם רק דרך האתר.
2. פעולות פשוטות הנועדו למציאת הטלפון כגון: השמעת קולות, שינוי הווליום (למקרה שהטלפון מושתק), מציאת מיקום בעזרת GPS, צילום תמונות (כדי לראות איפה הוא מנקודת המבט שלו) והפעלת ההבזק של המצלמה.
3. מנהל תהליכים: מציג את התהליכים, נותן להתחיל ולסגור תהליכים, שומר שמות תהליכים שרצים על הטלפון במסד הנתונים כדי שהמשתמש יוכל לפתוח אותם גם אם שכח את שמם, יוצר פרופיל שימוש של המשתמש ומתריע על תהליכים חדשים שאינם תואמים לפרופיל.
4. שמירת מידע על המכשיר בענן.
5. שיתוף קבצים עם משתמשים אחרים של השירות.

הערה - הפרוייקט נכתב ע"י תלמיד הלומד בתוכנית גבהים (יניב) ותלמיד הלומד שירותי רשת (יובל). המסמך הזה הוא מסמך עיצוב הפרוייקט של יניב ולכן יפרט רק את החלקים שיניב צריך לעשות בפרוייקט.

מסמכים קודמים:

מסמכי איפיון:

יניב - <https://docs.google.com/document/d/1V75tLstmSVEeeHeCAWPcI-QEpd6rMnnqJ3jDzwsxxso/edit?usp=sharing>

יובל - <https://docs.google.com/document/d/1Ta4LLAR_R8x6pwXa6LORB3QdExOFo6VaOf-che4mHGs/edit?usp=sharing>

הגדרות מושגים:

**ארכיטקטורת המערכת**

מבט על

המודולים באפליצייה למכשיר האנדרואיד:

* שכבת האינטרקציה עם מערכת ההפעלה - אחראית על כל הפעולות שדורשות התממשקות עם מערכת ההפעלה.
* שכבת ההפעלה - שכבה האחראית על ביצוע הפעולות המבוקשות ע"י השרת. משתמשת בשכבת האינטרקציה עם מערכת ההפעלה (כדי לבצע פעולות) ובשכבת הרשת (כדי לשלוח מידע חזרה לשרת) כאשר צריך.
* שכבת התצוגה - שכבה האחראית על ניהול הממשק הגרפי של האפליקציה וביצוע פעולות המבוקשות ע"י המשתמש (שמירת שם משתמש וסיסמא וניהול הגדרות התוכנה).
* שכבת הרשת והפיענוח - שכבה האחראית על התממשקות עם השרת. השכבה מנהלת את כל התקשורת עם שרת ה-proxy ואחראית על להוות ממשק לתוכנה לגישה לשרת.

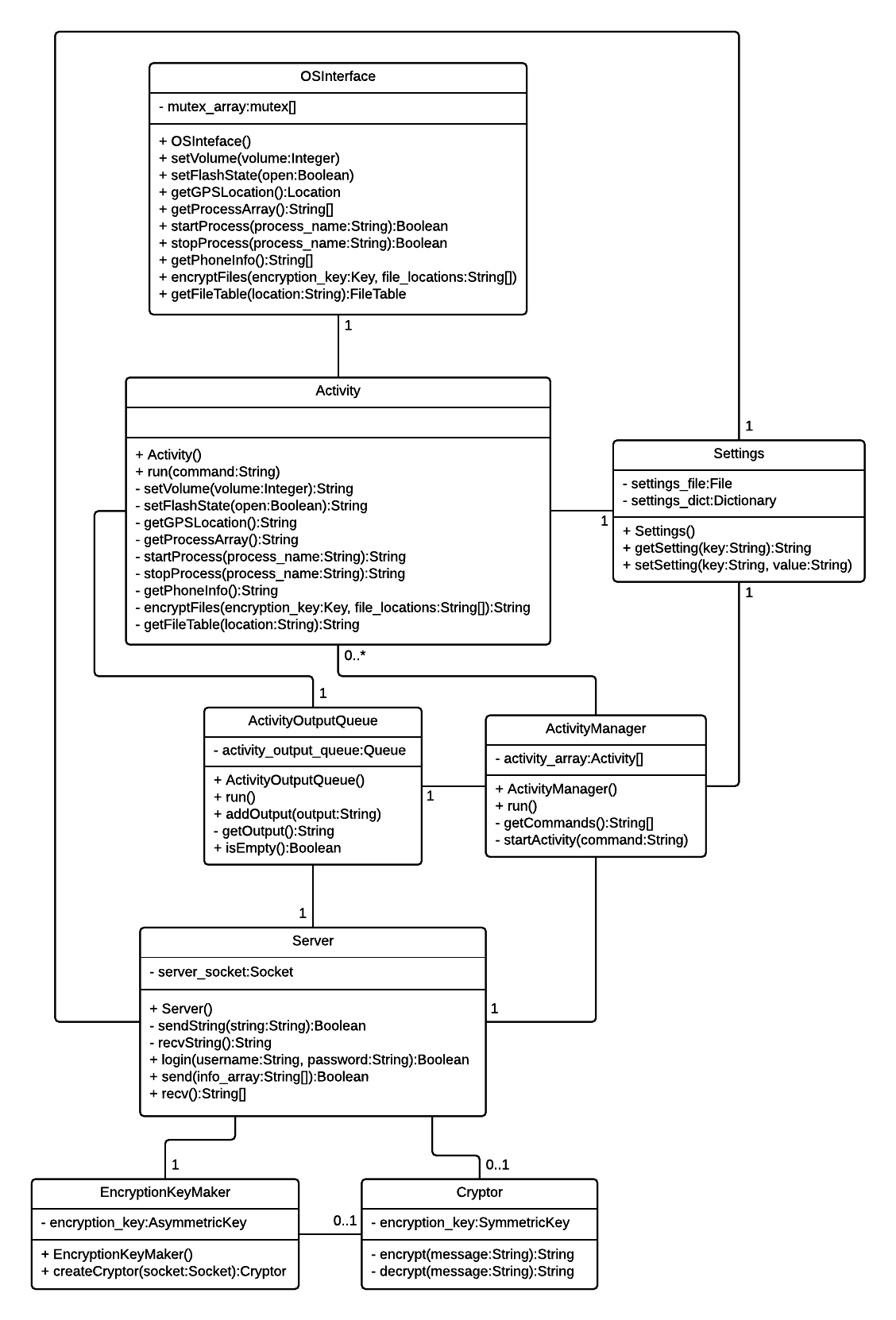
המודולים בשרת ה-proxy שבפייטון (Phone communication server):

* שכבת הרשת - שכבה האחראית על ניהול התקשורת בין השרת לשרת הראשי ולטלפונים.
* שכבת הסינון - בשכבה זו יסוננו ההודעות שמתקבלות משכבת הרשת שאינן תואמות לפרוטוקול והן לא יועברו לשרת הראשי.
* שכבת התצוגה - שכבה המציגה הודעות על פעולת המערכת. לשרת לא ממשק גרפי והוא ירוץ כ-console application. השרת ידפיס הודעות על הפעילות שלו.
* שכבת ההפעלה - אחראית על להפעיל את כל השרת. קוראת לכל האובייקטים.

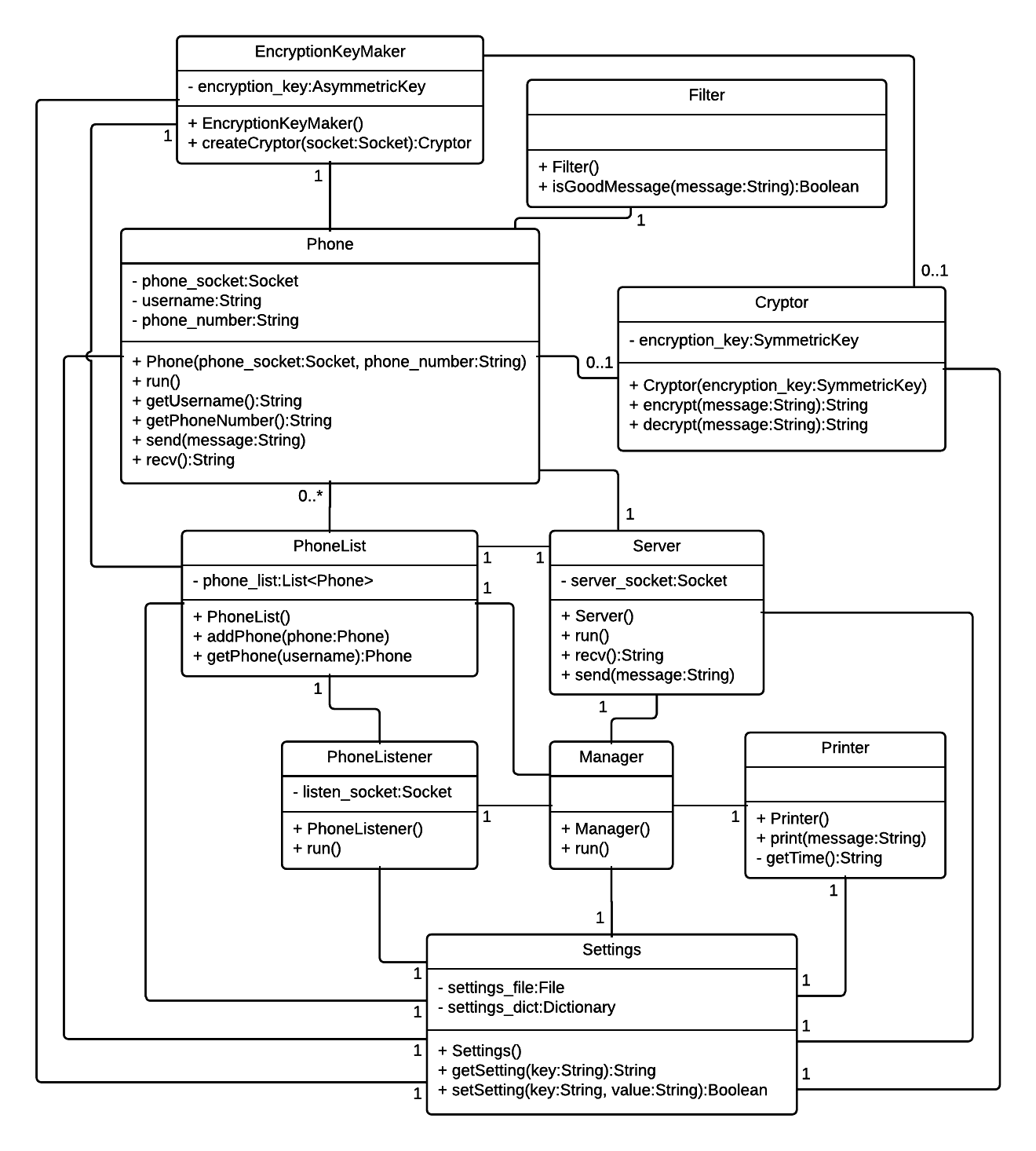
השרת הראשי הינו שרת הנכתב ע"י יובל והוא ינהל את התקשורת בין הלקוח למכשיר האנדרואיד שלו.

פירוט רכיבי המערכת

האפליקציה למכשיר האנדרואיד:

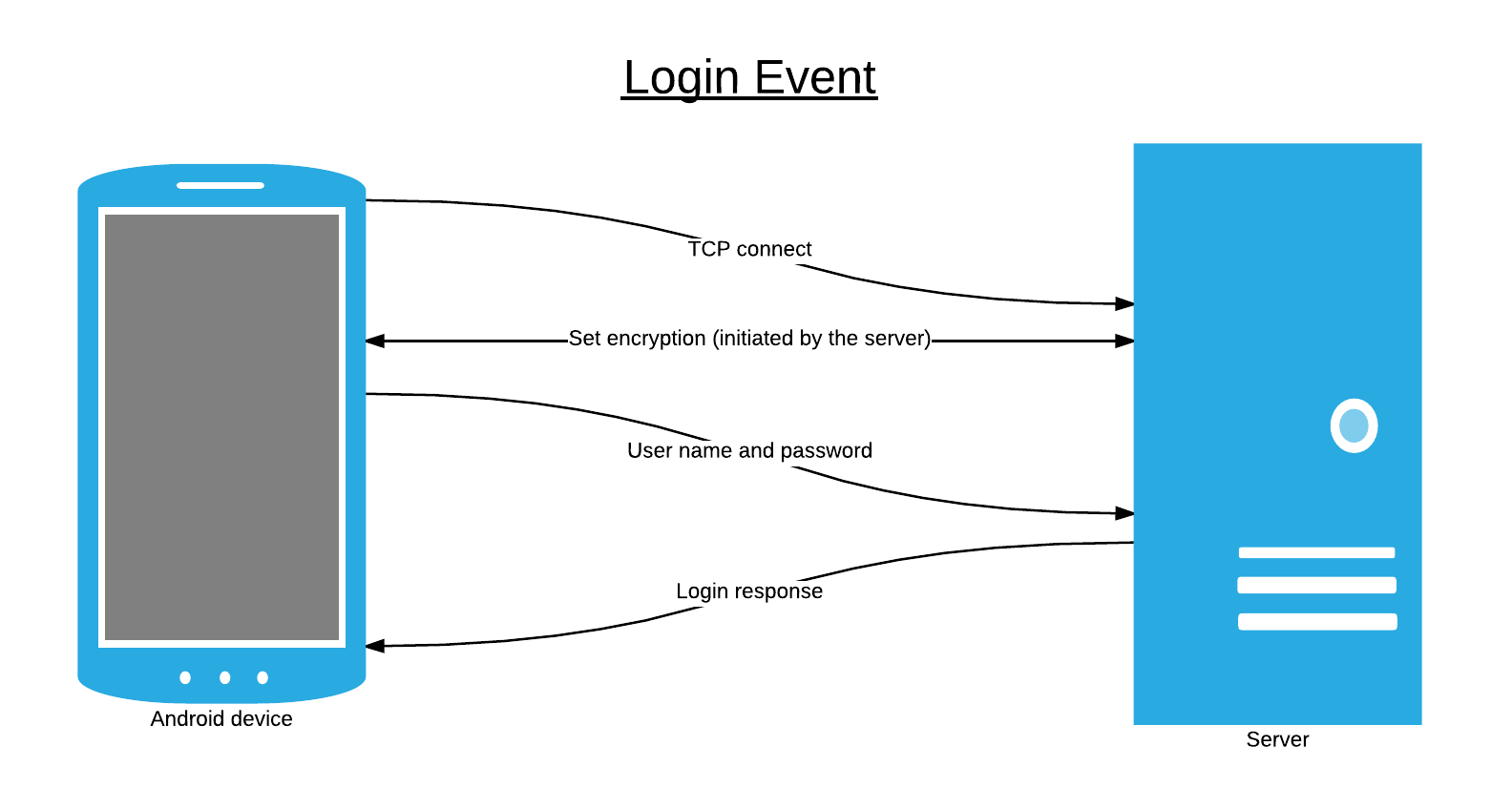


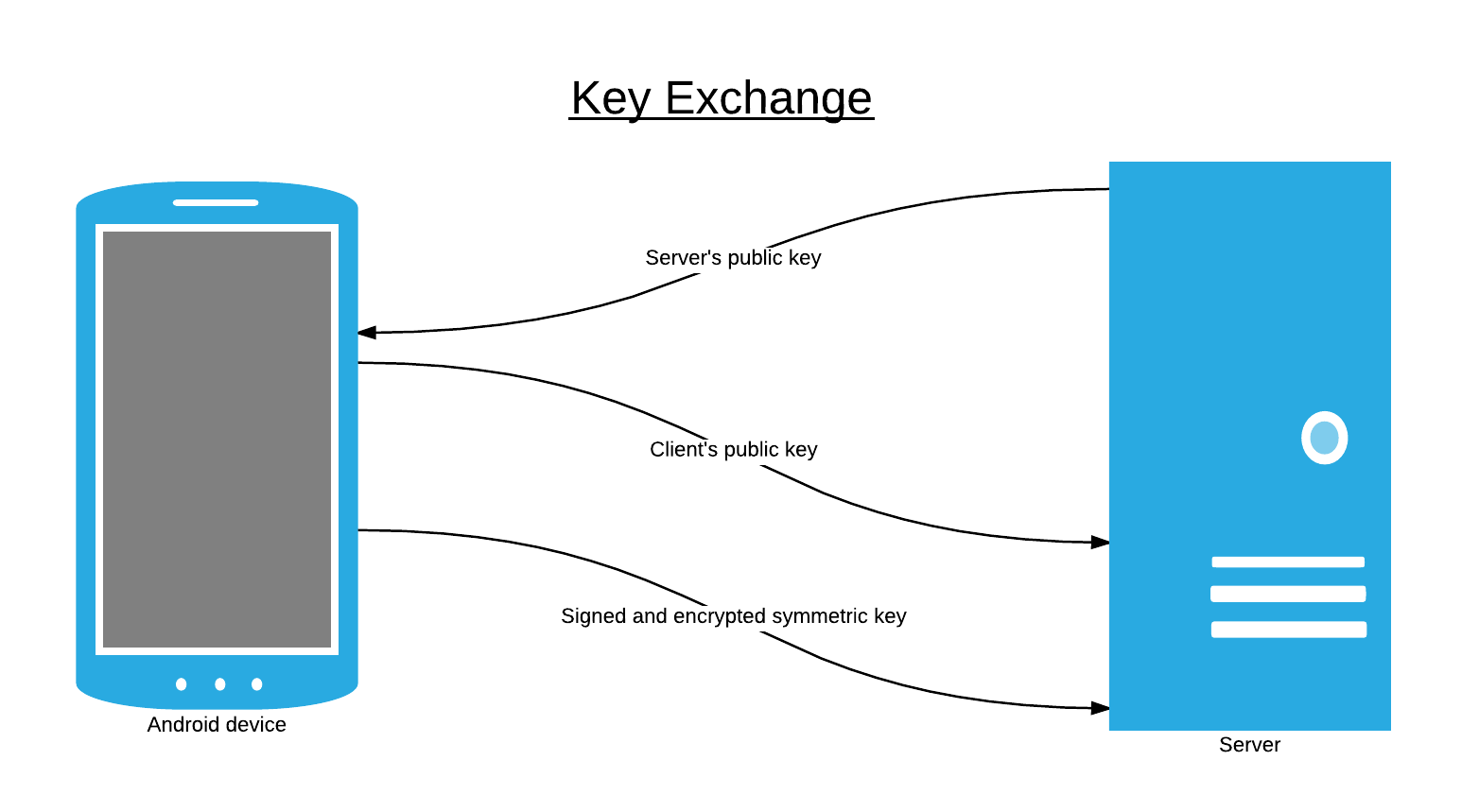
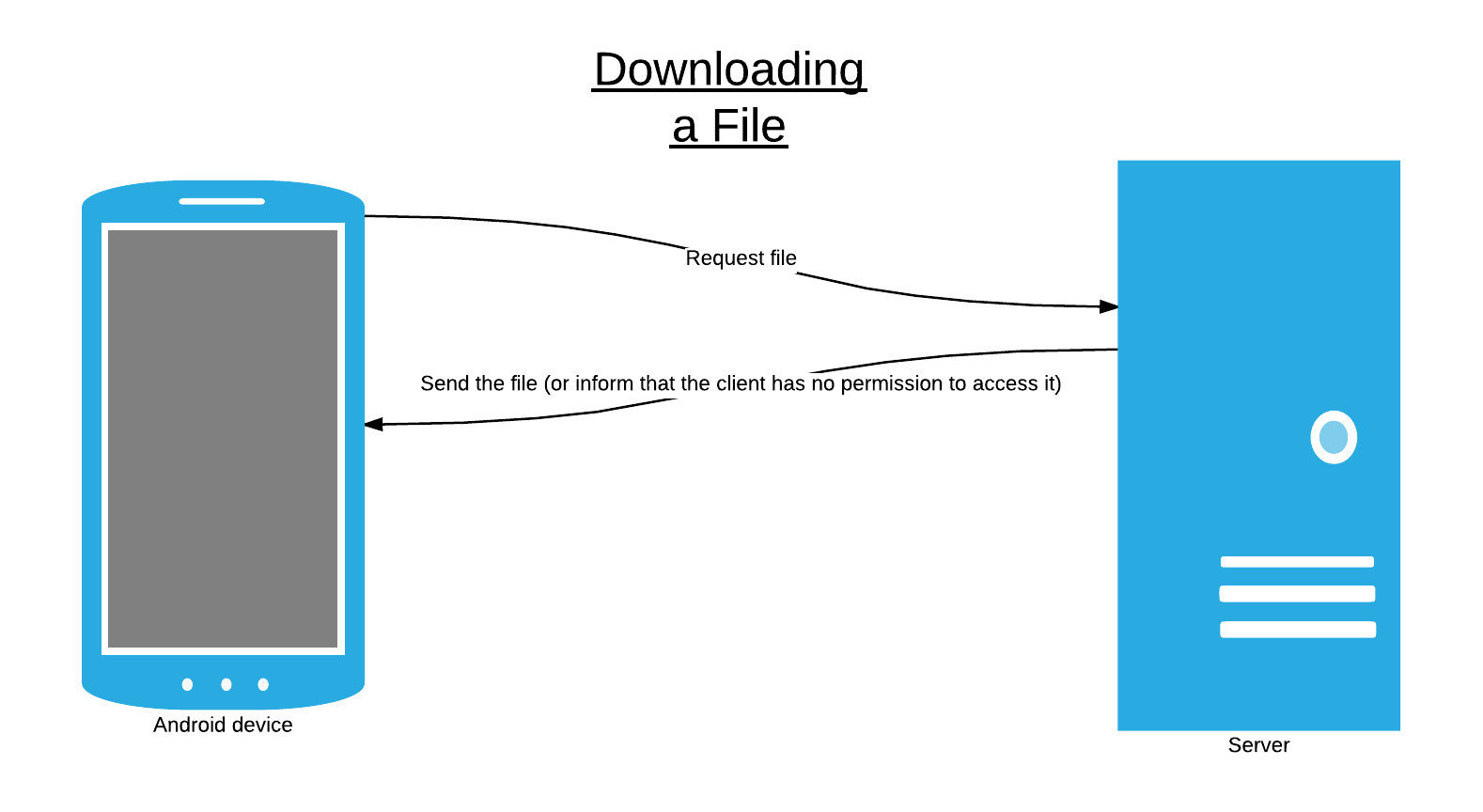
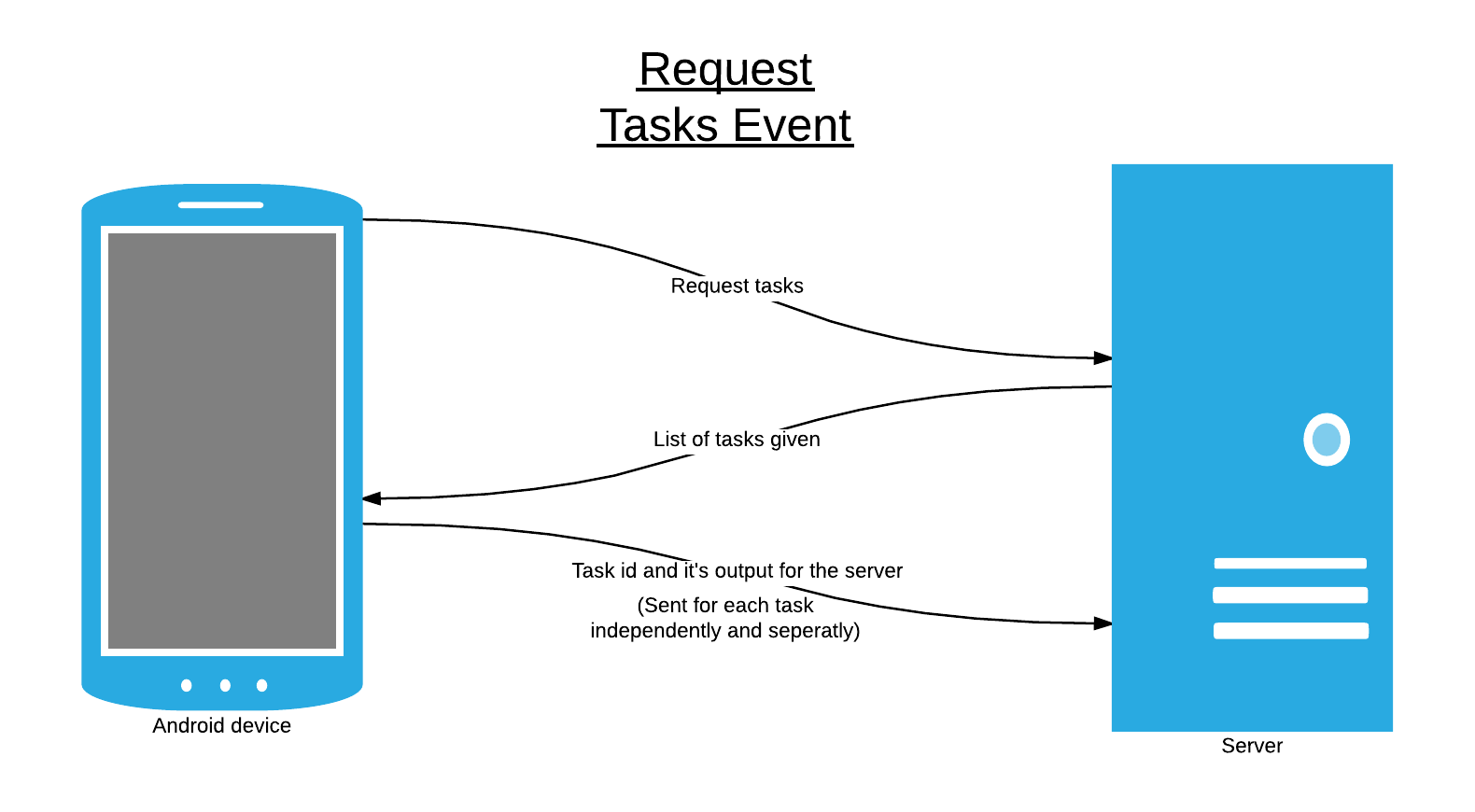
שרת הפרוקסי:



בחרתי לבנות את התוכנות בצורה הזאת מכיוון שבעיצוב זה התוכנות מאוד מודולריות וניתן לעשות multithreading בקלות. בשרת ובאפליקצייה ישנו thread ראשי (Manager ו- ActivityManager) והוא מנהל את כל שאר ה-threads. לשרת proxy בחרתי להשתמש ב-Python מכיוון שזו שפה נוחה וקלה לשימוש. לאפליקציית Android בחרתי להשתמש ב-Java מכיוון שגם היא שפה קלה וגם היא השפה שיש לה הכי הרבה תמיכה מצד גוגל (היא השפה שבה משתמשים ב-Android Studio שהוא הקומפיילר לאנדרואיד הרשמי של גוגל).

**נתונים ופרוטוקולים**





המבנה של הפקטות:

כל הפקטות של הלקוח מתחילות ב-

Rdroid CLIENT

כל הפקטות של שרת הפרוקסי מתחילות ב-

Rdroid PROXY

כל הפקטות של השרת הראשי מתחילות ב-

Rdroid SERVER

בסוף כל שורה בפקטה כתוב-

/r/n

בשורה השנייה של הפקטה כתובה מטרת הפקטה-

רשימת מטרות:

DOWNLOAD

ENCRYPTION

LOGIN

TASK

TASK\_RESULTS

NOTIFY\_SESSION\_ID

בשורה השלישית והלאה של הפקטה מצויינים הפרמטרים של הפעולה-

דוגמא של שורה שלישית ורביעית של פקטת LOGIN של הלקוח:

username: mighty

password: python

פרמטרים של פעולות:

DOWNLOAD (CLIENT):

requests a file to download from the server.

location - the location of the requested file.

ENCRYPTION:

used to send an encryption key to the other agent.

key - the key that is sent.

LOGIN (CLIENT):

logins into the server.

username - the username of the client

password - the password of the client

LOGIN (SERVER):

informs the client whether the login was successful or not.

result - answers whether the login was successful or not.

TASK (CLIENT):

requests the current list of tasks from the server.

exclude - a list of task ids that the client already knows about but haven't finished.

TASK (SERVER):

sends a list of task that the client needs to do

task block - because this packet can send more than one task, each task will be defined in a block where all of its parameters are present.

<Task> - indicates the start of a task block.

</Task> - indicates the end of a task block.

id - the id of the task used to identify the results that are sent back to the server after the task was done.

type - the type of the task.

parameter - used when the task needs parameters. example: can provide a file address that is used by the client to request a file to download.

TASK\_RESULTS (CLIENT):

sends the results of the tasks that were given to the client.

task\_id - the id of the task.

output - the results of the task or the information that it needs to return.

NOTIFY\_SESSION\_ID (PROXY)

tells the server of a new session id that is given to a phone by the proxy.

session\_id - the id of the session

בנוסף, במעבר בין השרת הראשי לשרת הפרוקסי ובין שרת הפרוקסי לשרת הראשי כל פקטה תקבל בתחילתה את ה-id של ה-session. לדוגמא (id=123):

123:message

הנקודותיים מסמלות את סוף ה-id ותחילת ההודעה עצמה.

מבנה שמירת ההגדרות של האפליקציה

ההגדרות של האפליקציה יישמרו במסמך טקסטואלי. ההגדרות יישמרו בצורה של מילון: כל הגדרה תשמר כמפתח וערכו. כל שורה במסמך תהיה הגדרה נפרדת. בכל שורה, המפתח הוא הטקסט עד ה-= וערכו הוא כל שאר הטקסט. שורות המתחילות ב-# הינן שורות הערה ולכן אינן נקלטות כהגדרות בתוכנה.

דוגמא:

#network

server\_ip=133.74.20.42

server\_port=8888

#account

name=mighty

password=python

במקרה הזה שמור שכתובת השרת היא 133.74.20.42:8888 שם המשתמש הוא mighty והסיסמא היא python.

**ממשק המשתמש**

לשרת ה-proxy יהיה ממשק של console ולכן לא אפרט עליו. לאפליקציה יש 4 פעילויות (המקביל ל-forms באנדרואיד):

login\_activity

האחראית על התחברות לשרת עם המשתמש

main\_activity

האחראית על בחירה לאיזו מבין שתי הפעילויות הבאות לעבור

settings\_activity

האחראית על עריכת הגדרות האפליקציה

encrypted\_containers\_activity

האחראית על פתיחת הקונטיינרים המוצפנים