

پروژه درس اینترنت اشیا

کمد هوشمند فردوسی

استاد درس: دکتر محسن کاهانی

دیبا امین شهیدی

حجت الله سرگلزهی

هما حبشی

آرمان قرشی

مبین وحدتی ترکها

فاز سوم

بهار ۹۹

چکیده

با پیشرفت علم، علم اینترنت اشیا نیز در چند دهه اخیر بسیار مورد توجه قرار گرفته. در درس اینترنت اشیا، تیم ما پروژه ای را تعریف کرده تحت عنوان کمد هوشمند فردوسی. در این پروژه هدف هوشمند سازی پروسه گرفتن کمد در مکان های عمومی مانند فضای دانشگاه بوده. این کار باعث استفاده بهینه از منابع موجود در دانشگاه با توجه به بیشتر بودن عرضه نسبت به تقاضا می شود. همچنین کار نیروی انسانی را کمتر کرده و خطا کمتر می شود و کاربر تجربه سریع و امن بودن را خواهد داشت. در این پروژه یک اپ اندروید برای کاربران طراحی شده که به وسیله آن می توانند کمد بگیرند و آن را مدیریت کنند. ادمین سیستم با استفاده از پنل وب ای که در اختیار او قرار می گیرد می تواند کل سیستم را مدیریت و مانیتور کند. وظیفه پردازش دستورات و محاسبات و ذخیره اطلاعات بر عهده سرور می باشد که میان اپ و سخت افزار کنترل کننده کمد ها قرار می گیرد.

• مقدمه	۴
• نمودار UML	۶
• آموزش کار با سیستم	۸
○ کاربر نهایی	۸
○ ادمین	۱۴
• جزئیات فنی	۱۸
○ معماری سخت افزار	۱۸
○ معماری سرور	۲۱
▪ api ها	۲۳
○ رابط کاربری	۲۷

۱. مقدمه

هرروزه شاهد هوشمند شدن بخش های مختلفی از زندگی خود هستیم. مکان هایی مانند محل کار، پارکینگ ها ، خانه ها و ... با پیشرفت تکنولوژی دچار تغییرا هایی می شوند. این هوشمند شدن کار ها را ساده تر، سریع تر و دقیق تر می کند.

در این پروژه، هدف هوشمند سازی کمد هایی است که در مکان های عمومی قرار می گیرند. این هوشمند سازی باعث می شود مدیریت این کمد راحت تر شود. همچنین تعامل کاربران را نیز ساده تر می کند.

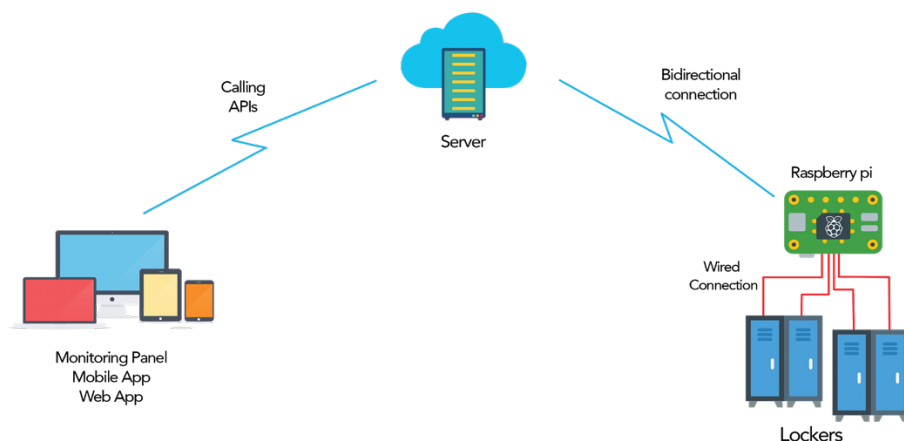
این کمد های هوشمند را می توان در مکان های مختلف برای استفاده عموم قرار داد.

- **اماکن ورزشی:** در سرتاسر شهر مشهد امکان ورزشی عمومی زیادی وجود دارند که افراد بسیار از آن ها استفاده می کنند و هنگام استفاده وسایل خود را در کنار زمین قرار می دهند و این کار امنیت کافی را ندارد و ممکن است سارق وسایل را بردارد. با وجود این کمد ها افراد می توانند برای همان مدتی که در آن جا مشغول فعالیت اند کمد اجاره کرده و با آرامش خاطر به فعالیت بپردازند.
- **مراکز خرید:** با قرار دادن این کمد ها در مراکز خرید افراد می توانند بجای حمل خرید های خود تا پایان زمانی که در مجموعه هستند، آن ها را در کمد قرار داده و با راحتی به گشت و گذار خود بپردازند
- **حرم:** افرادی که از شهر های دیگر به حرم می آیند گاها وسایلی همراه خود دارند که حمل آن هنگام زیارت و جابه جا شدن در محیط حرم برای آن ها دشوار است. وجود این کمد ها کار را برای افراد ساده می کند.
- **محیط دانشگاه:** کمد هایی که در دانشگاه وجود دارد معمولا برای مدت طولانی دست دانشجویان می ماند و ممکن است اکثر اوقات بدون استفاده باشد. وجود این کمد ها در دانشگاه این مشکل را برطرف کرده و در همان مدت هایی که کمد بلا استفاده بود ، افراد دیگری قادر به استفاده از آن خواهند شد.
- **فرودگاه :** افراد در فرودگاه ها معمولا وسایل زیادی همراه خود دارند و ممکن است تا زمانی که نوبت به پرواز آن ها برسد بخواهند در محیط فرودگاه گشت و گذاری داشته باشند که حمل این وسایل کار را دشوار می کند. وجود کمد هایی با اندازه های مناسب این وسایل می توان برای مسافران رفاه بیشتری فراهم کند.

نمونه های مشابهی از این کمد ها در فرودگاه مالزی، بازار های دبی و دانشگاه های مختلف مورد استفاده قرار می گیرد.

بزرگترین مزیت هوشمند سازی کمد ها، حل مشکل مدیریت و نگهداری از کلید برای کاربران ما است. این هوشمند سازی کلید و مشکلات حاصل از آن را حذف می کند. همچنین مدیریت کمد ها باید توسط یک فرد انجام می شد و کلید ها را به کاربران می داد و بازپسگیری آن را پیگیری می کرد اما با حذف کلید این بخش دشوار و مشکلات آن نیز حذف می شود. از مزایای دیگر این هوشمند سازی، ظاهر به روز محیط و قدرت تحلیل و مانیتور سیستم و کاربران است.

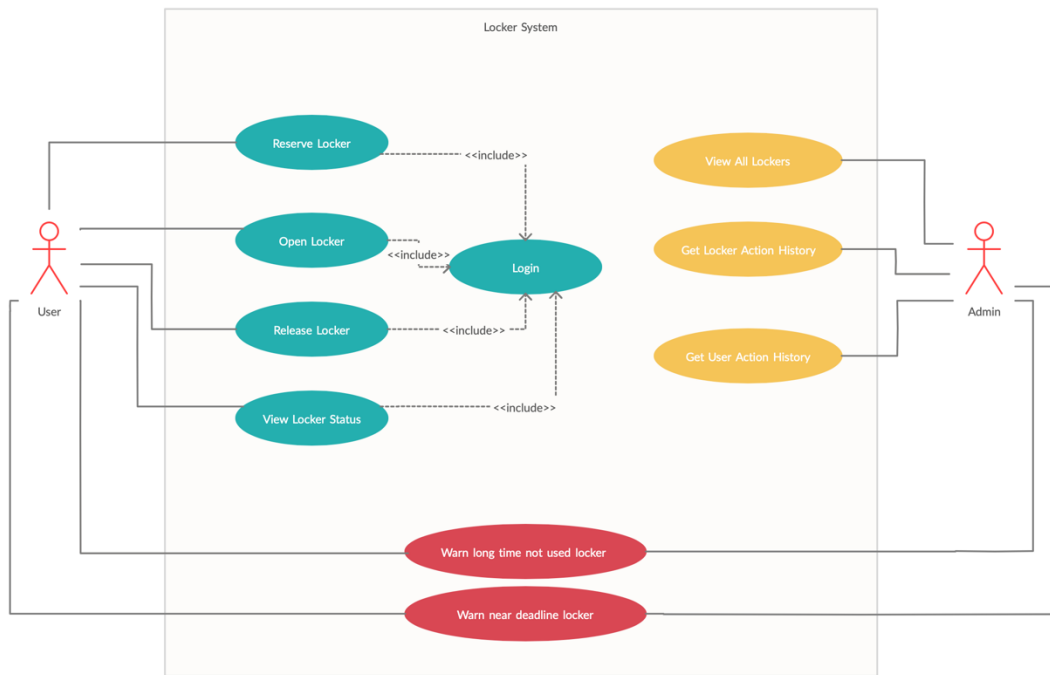
معماری سیستم در یک نگاه کلی و جامع، به جهت ایجاد تصور اولیه از نحوه کار آن، مانند تصویر زیر است:



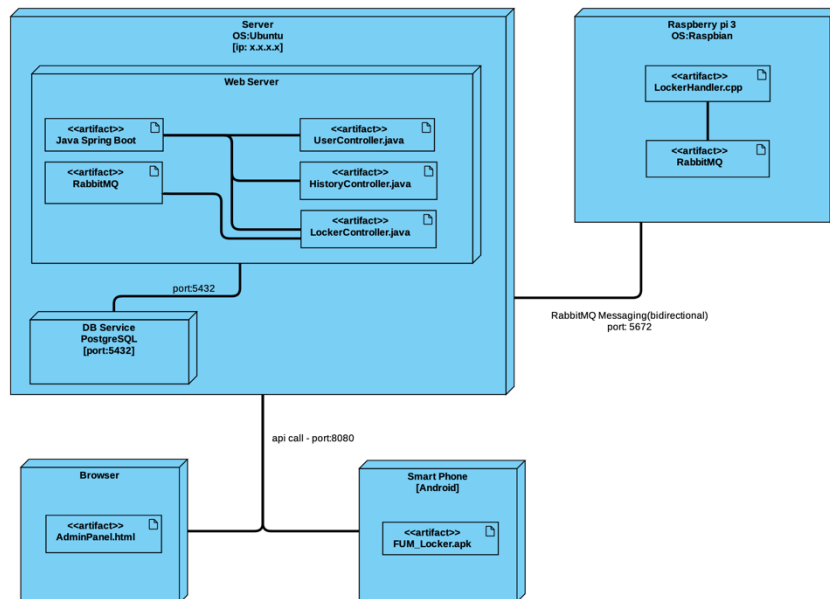
همانطور که قابل مشاهده است این سیستم به سه بخش کلی تقسیم می شود. سخت افزار، سمت سرور، سمت کاربر (کلاينت)

- در بخش سخت افزار از رزبری پای به عنوان قلب پردازشی استفاده می شود که وظیفه کنترل سخت افزاری کمد ها را بر عهده دارد و دستورات را از سرور در قالب یک ارتباط دو طرفه می گیرد.
 - در بخش سرور، اطلاعات کاربران و کمد ها ذخیره شده و پردازش دستور ها در این قسمت انجام می شود.
 - سمت کاربر نیز دارای رابط هایی مانند اپلیکیشن اندروید و پنل وب است تا کاربر بتواند به ساده ترین و سریع ترین روش با سیستم به تعامل بپردازد و ادمین سیستم نیز بتوان سیستم را کنترل و مانیتور کند.
- جزئیات فنی سیستم در بخش ۳ توضیح داده شده است.
- جهت مشاهده فیلمی از دمو ی این سیستم بر روی لینک مقابل کلیک کنید.

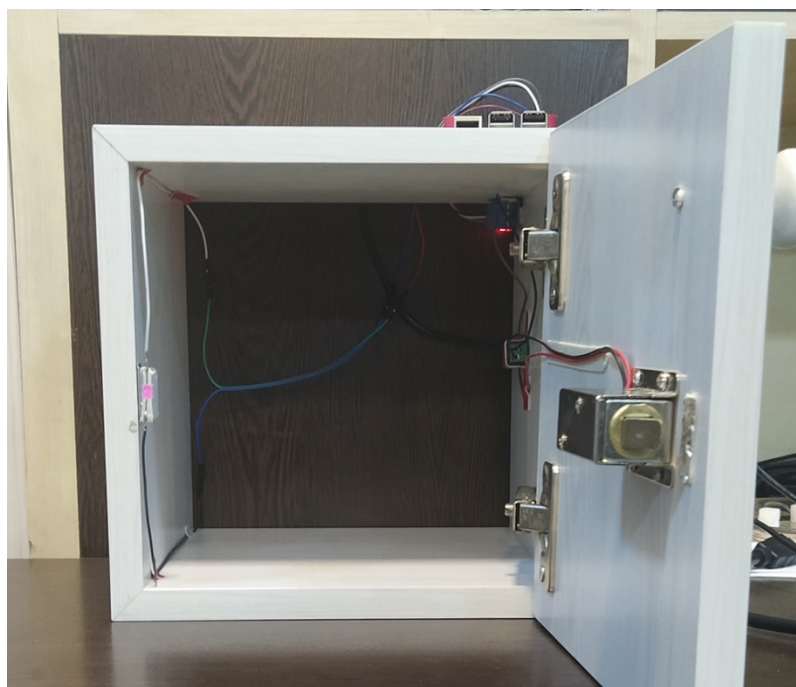
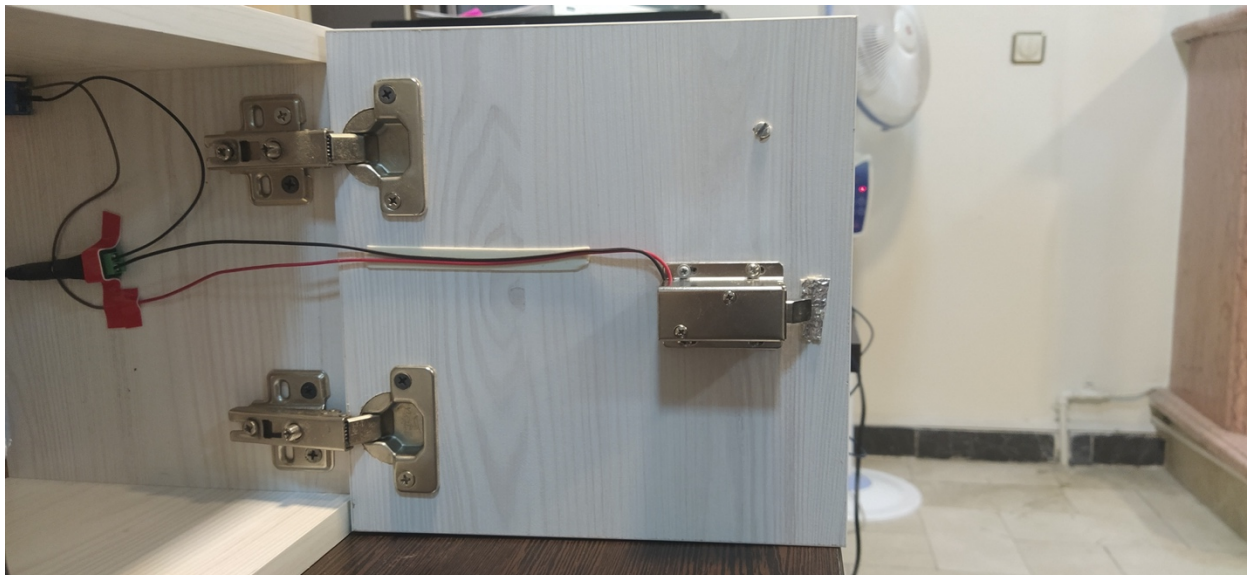
در زیر نمودارهای UML آمده است.



نمودار usecase



نمودار deployment



تصاویری از کمد

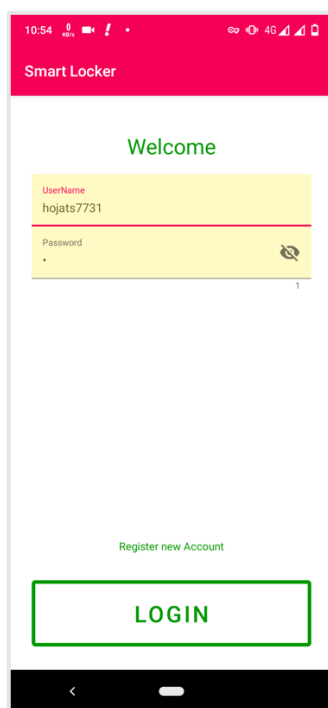
۲. آموزش کار با سیستم

در این بخش از مستند قصد داریم چگونگی تعامل افراد با سیستم را شرح دهیم. این کار را می توان از دو جهت بررسی کرد. تعامل ادمین با سیستم و تعامل کاربر نهایی با سیستم.

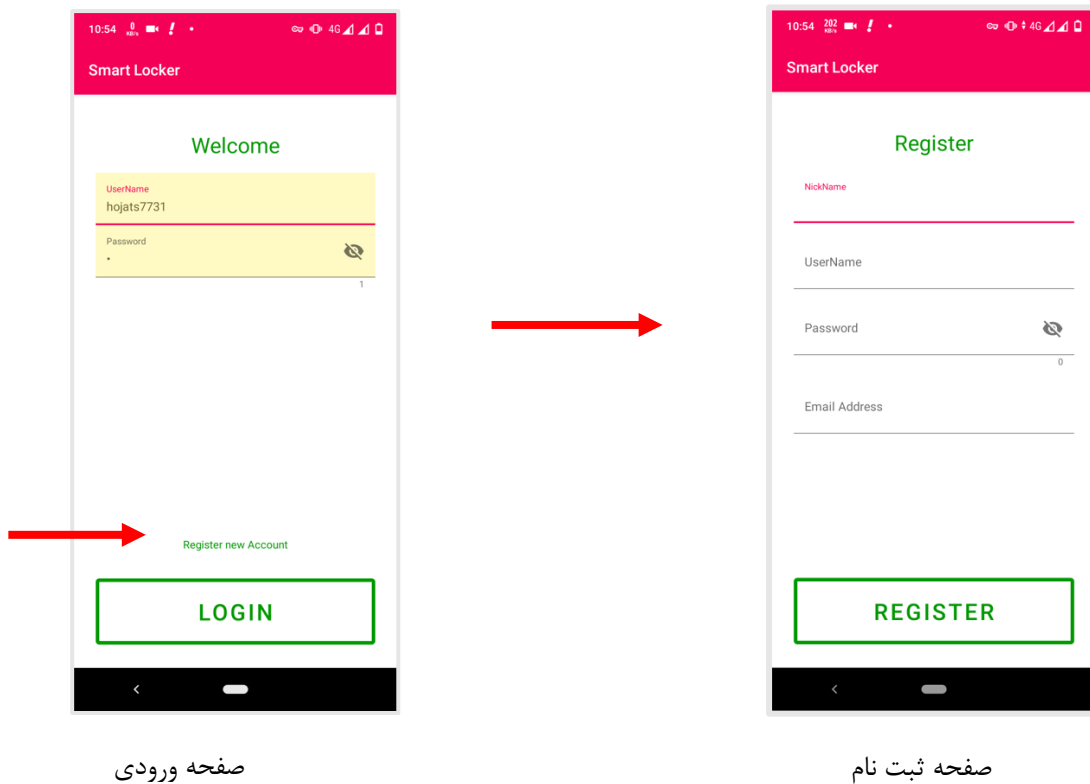
۲.۱. کاربر نهایی

کاربران برای بهره مندی از امکانات کمد های هوشمند فردوسی باید مراحل زیر را اجرا کنند.

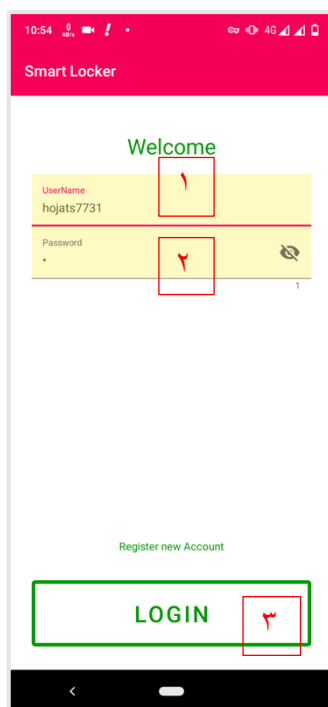
- ابتدا اپلیکیشن اندروید مربوط به کمد های هوشمند فردوسی را بر روی تلفن هوشمند خود نصب کنید. پس از نصب و باز کردن اپ با صفحه زیر مواجه می شوید.



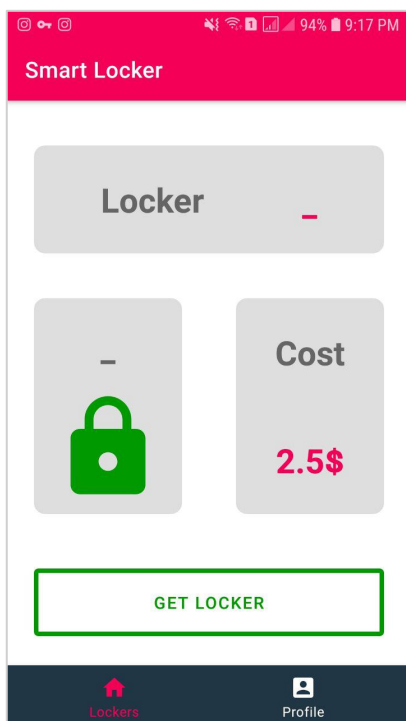
- در صورتی که از پیش حساب کاربری ندارید باید در سیستم ثبت نام کنید. بر روی دکمه ثبت نام بزنید و اطلاعات را وارد کنید.



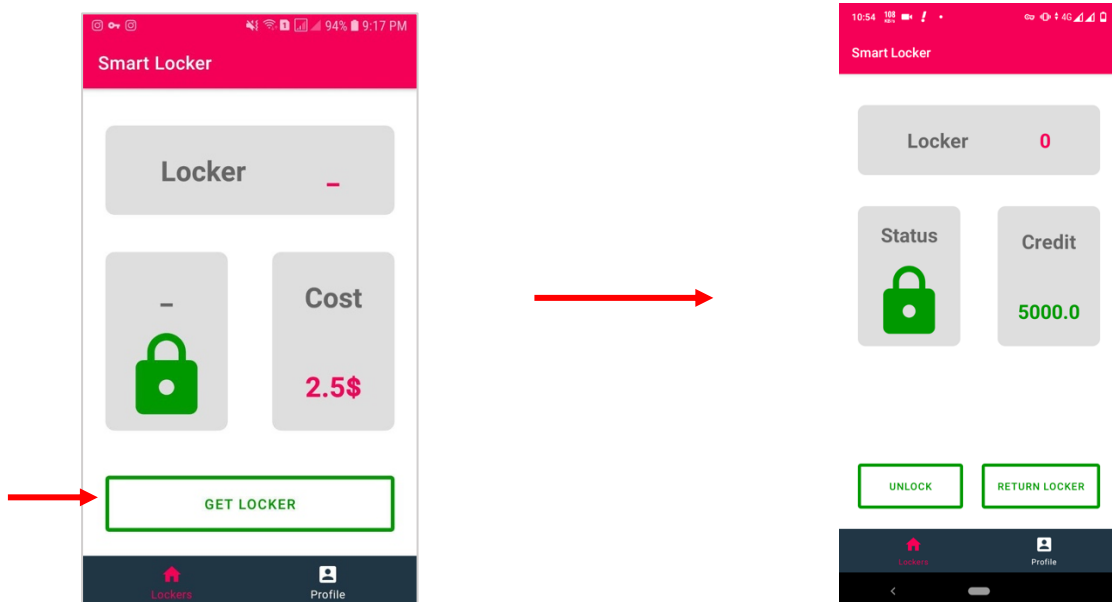
- پس از این که حساب کاربری خود را بدست آوردید یا اگر از پیش داشتید، برای استفاده از امکانات اپ ، وارد حساب خود شوید.



- پس از این که وارد شدید با صفحه زیر مواجه می شوید. هنگامی که هنوز کمدی در اختیار شما قرار نگرفته ، قسمت مربوط به شماره کمد خط تیره خواهد بود.

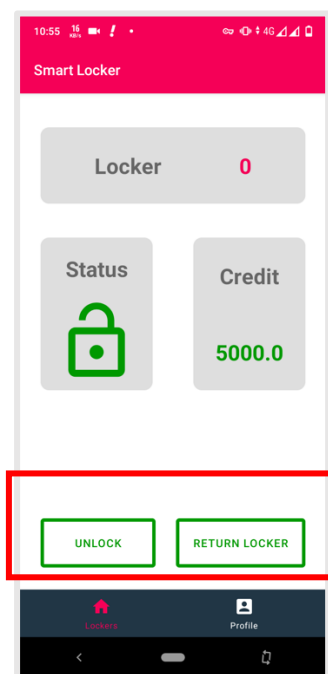


- اگر کمد ندارید بر روی دکمه Get Locker بزنید تا از سوی سرور یک کمد به شما اختصاص داده شود. علامت قفل ای که در وسط صفحه سمت چپ وجود دارد نشان دهنده وضعیت کمد (باز یا بسته) است.



همانطور که مشاهده می شود شماره کمد و وضعیت آن، پس از دریافت کمد، در صفحه نمایش داده می شود.

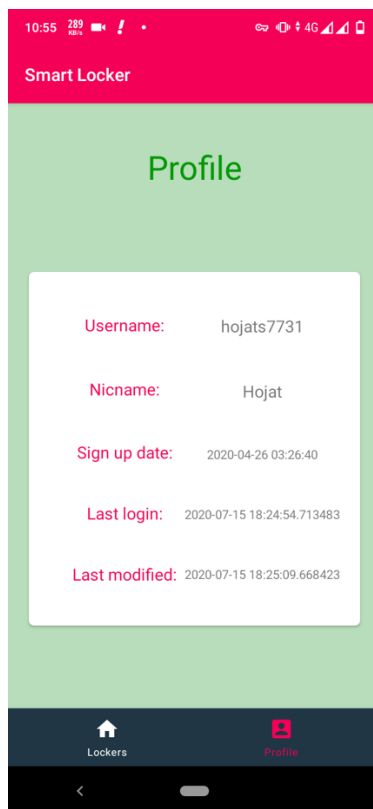
- پس از گرفتن کمد، دو دکمه در صفحه ظاهر می شود (در شکل با مستطیل قرمز مشخص شده) :
Unlock : برای باز کردن در کمد استفاده می شود.
Return Locker : برای پس دادن کمد مورد استفاده قرار می گیرد.
کاربر برای استفاده از کمد به جای کلید، از دکمه Unlcok برای باز کردن در کمد استفاده می کند.
پس از باز شدن کمد وضعیت قفل به صورت زیر در می آید.



- پس از اتمام کار کاربر با کمد، باید در آن کمد را به داخل بفشارد تا قفل در جای خود جا بیفتد و در بسته شود. اگر در به درستی بسته شود وضعیت کمد در تلفن هوشمند کاربر تغییر کرده و به حالت بسته می رود.
- هنگامی که کاربر دیگر قصد استفاده از کمد را ندارد می تواند با زدن دکمه Return Locker ، پس از برداشتن وسایل خود و بستن در، اقدام به پس دادن کمد کند.
- برای استفاده از کمد یک سیستم پرداخت در نظر گرفته شده که می تواند منبع درآمدی برای شهرداری، دانشگاه و یا هر کسی که این کمد ها را راه اندازی کرده باشد. به این صورت که شخص برای

استفاده از کمد ها باید حساب خود را شارژ کند. استفاده از کمد هزینه روزانه دارد که پایان هر روز از حساب کسر می شود. در صورتی که میزان حساب شخص کافی نباشد یک هشدار برای او و ادمین ایمیل می شود. در صورتی که فرد حساب خود را شارژ نکند یا کمد را پس ندهد، نمی تواند در کمد را باز کند تا زمانی که هزینه منفی حساب خود را صفر کند.

- در پایین اپ یک سربرگ دیگر به نام Profile وجود دارد که در آن اطلاعات کاربر نمایش داده می شود.



- برای استفاده از پنل کاربری وب ابتدا آدرس زیر را در مرورگر وارد کنید تا صفحه زیر بارگزاری شود.
<http://5.160.146.108/IoT/Login/login.html>

The image shows the FUM Smart Locker web interface. At the top is a header with the text "FUM Smart Locker". Below the header are two side-by-side forms. The left form is titled "Login" and contains fields for "Username" and "Your password" (masked with asterisks), followed by a blue "Login" button. The right form is titled "Sign Up" and contains fields for "Username", "Nickname", "Email", and "Choose Password" (masked with asterisks), followed by a blue "Sign Up" button.

- اگر حساب کاربری ندارید در سمت راست یک حساب بسازید و در سمت چپ وارد شوید.

The image shows the FUM Smart Locker dashboard after a successful login. The browser address bar shows the URL: `File | /Users/dibaaminshahidi/Desktop/Work/Spring99/IOT/Project/Web-UI-/User/Dashboard.html`. The dashboard has a dark header with a "Sign Out" button. Below the header, there are two main sections. On the left, under "Locker ID", it shows the number "8" and a large padlock icon. Below the icon are two blue buttons: "Unlock" and "Return Locker". On the right, under "Account", there is a list of user details in grey boxes: "Username : yavar", "Nickname : yav", "Email : null", "Sign Up At : 2020-07-15T13:45:21.272635", "Last Login At : 2020-07-16T01:56:03.037035", and "Credit : 4900".

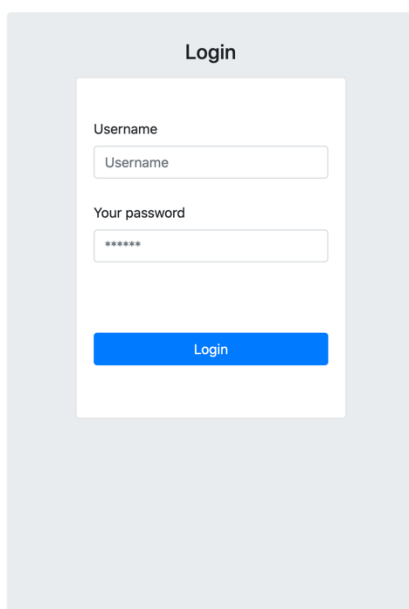
پس از ورود در این صفحه می تواند از امکانات مانند اپ اندروید بهره مند شوید. این صفحه با موبایل هم همخوانی دارد.

۲.۲ کاربر ادمین

مدیر یا همان ادمین سیستم برای استفاده از امکانات پنل مدیریت می تواند مراحل زیر را طی کند

- ابتدا باید آدرس پنل ادمین را در مرورگر وارد کند و در سیستم Login کند.

<http://5.160.146.108/IoT/Login/login.html>



یوزر و پسورد پیشفرض برای کاربر ادمین:

User: admin

Pass: abcd1234

*توجه شود که a یوزرنیم باید کوچک باشد.

پس از ورود به پنل مدیریت امکانات زیر برای او فراهم شده است که هرکدام را توضیح می دهیم.

- در صفحه اصلی دو جدول وجود دارد.

یک جدول از تمامی کمد ها و اطلاعات مربوط به آن ها شامل موارد زیر نمایش داده می شود :

- آیدی کمد

- باز یا بسته بودن هر کمد

- مالک کمد در صورت وجود

یک جدول دیگر از هشدار ها وجود دارد که شامل هشدار های زیر است:

- باز شدن در کمد به صورت غیر مجاز و غیر از دستور صاحب کمد.
- کافی نبودن اعتبار کاربر برای گرفتن کمد در روز های آینده.
- بلا استفاده بودن یک کمد که در اختیار یک کاربر است برای یک مدت طولانی.

البته علاوه بر نمایش این هشدار ها در این قسمت ، برای کاربر و ادمین نیز ایمیل می شود.

FUM Smart Locker

Admin Panel

Menu

Overall Status

ID	Status	Owner
7	CLOSE	mvtkm
6	CLOSE	dib
8	CLOSE	NOT RESERVED
3	CLOSE	homz
0	CLOSE	hojats7731

Warnings

ID	Message	Email	Username	Time
0	UNEXPECTED_OPENING	hojats7731	hojats7731@gmail.com	2020-07-15T15:18:3
0	UNEXPECTED_OPENING	hojats7731	hojats7731@gmail.com	2020-07-15T15:18:3
0	UNEXPECTED_OPENING	hojats7731	hojats7731@gmail.com	2020-07-15T15:18:3
0	UNEXPECTED_OPENING	hojats7731	hojats7731@gmail.com	2020-07-15T15:18:3
0	UNEXPECTED_OPENING	hojats7731	hojats7731@gmail.com	2020-07-15T15:18:3
0	UNEXPECTED_OPENING	hojats7731	hojats7731@gmail.com	2020-07-15T15:18:3
0	UNEXPECTED_OPENING	hojats7731	hojats7731@gmail.com	2020-07-15T15:18:3

صفحه اصلی پنل ادمین شامل اطلاعات تمامی کمد ها و هشدار ها

- در بخش دیگری از پنل ادمین امکان گزارش گیری بر اساس کمد یا کاربر وجود دارد. برای این کار بر روی گزینه Menu که به رنگ آبی در بالا و سمت چپ صفحه وجود دارد کلیک کنید. سپس از بین گزینه ها بر روی Log بزنید.

FUM Smart Locker

Admin Panel

Menu

Dashboard

Logs

Sign Out

Overall Status

ID	Status	Owner
7	CLOSE	mvtkm
6	CLOSE	dib

Warnings

ID	Message	Email	Username	Time
0	UNEXPECTED_OPENING	hojats7731	hojats7731@gmail.com	2020-07-15T15:18:3
0	UNEXPECTED_OPENING	hojats7731	hojats7731@gmail.com	2020-07-15T15:18:3

FUM Smart Locker

Admin Panel

User Log

USER ID

Search!

گزارش تاریخچه اعمال کاربر

Locker History

LOCKER ID

Search!

گزارش تاریخچه کمد

Open Locker

LOCKER ID

Open!

صفحه Log پنل ادمین

در این بخش با وارد کردن نام کاربری کاربر مورد نظر و یا آیدی کمد مورد نظر و زدن دکمه Search ، یک تاریخچه از اعمالی که کاربر انجام داده یا اعمالی که روی یک کمد انجام شده به همراه زمان آن و اطلاعات دیگر نمایش داده می شود. مانند تصاویر زیر:

FUM Smart Locker

Admin Panel

Username : mobin

User ID : 19

Email : mvtikm@gmail.com

Locker ID	Action	Time
<>		
0	TAKE	2020-07-15T17:19:51.598627
0	CLOSE	2020-07-15T17:20:01.171423
0	OPEN	2020-07-15T17:19:57.048103
0	CLOSE	2020-07-15T17:19:59.092971
0	OPEN	2020-07-15T17:19:59.127383

User Log

mobin

Search!

Locker History

LOCKER ID

Search!

Open Locker

LOCKER ID

Open!

گزارش اعمال کاربر

FUM Smart Locker

Admin Panel

Menu

Locker ID : 7
Current Owner : mvtikm

Action	User ID	User Name	Time
<			
OPEN	1	mvtikm	2020-07-14T07:35:44.6962
OPEN	1	mvtikm	2020-07-14T07:37:06.232691
OPEN	1	mvtikm	2020-07-15T11:14:41.997184
OPEN	1	mvtikm	2020-07-15T11:16:23.828577
OPEN	1	mvtikm	2020-07-15T11:16:35.473895
OPEN	1	mvtikm	2020-07-15T12:48:30.585472
OPEN	1	mvtikm	2020-07-15T13:06:47.993081

User Log

Locker History

Open Locker

گزارش تاریخچه یک کمد

- در پنل ادمین این امکان وجود دارد که در صورت بروز مشکل برای اپ کاربر، بتوان از پنل ادمین در کمد را باز کرد. برای این کار در بخش مشخص شده در تصویر باید آیدی کمد مورد نظر وارد شود سپس دکمه open فشرده شود.

FUM Smart Locker

Admin Panel

Menu

User Log

Locker History

Open Locker

باز کردن اضطراری در کمد

بخش باز کردن در کمد ها در پنل ادمین

۳. جزییات فنی سیستم

همانطور که در مقدمه گفته شد بخش های اصلی سیستم را می توان سه مورد دانست:

سمت سرور

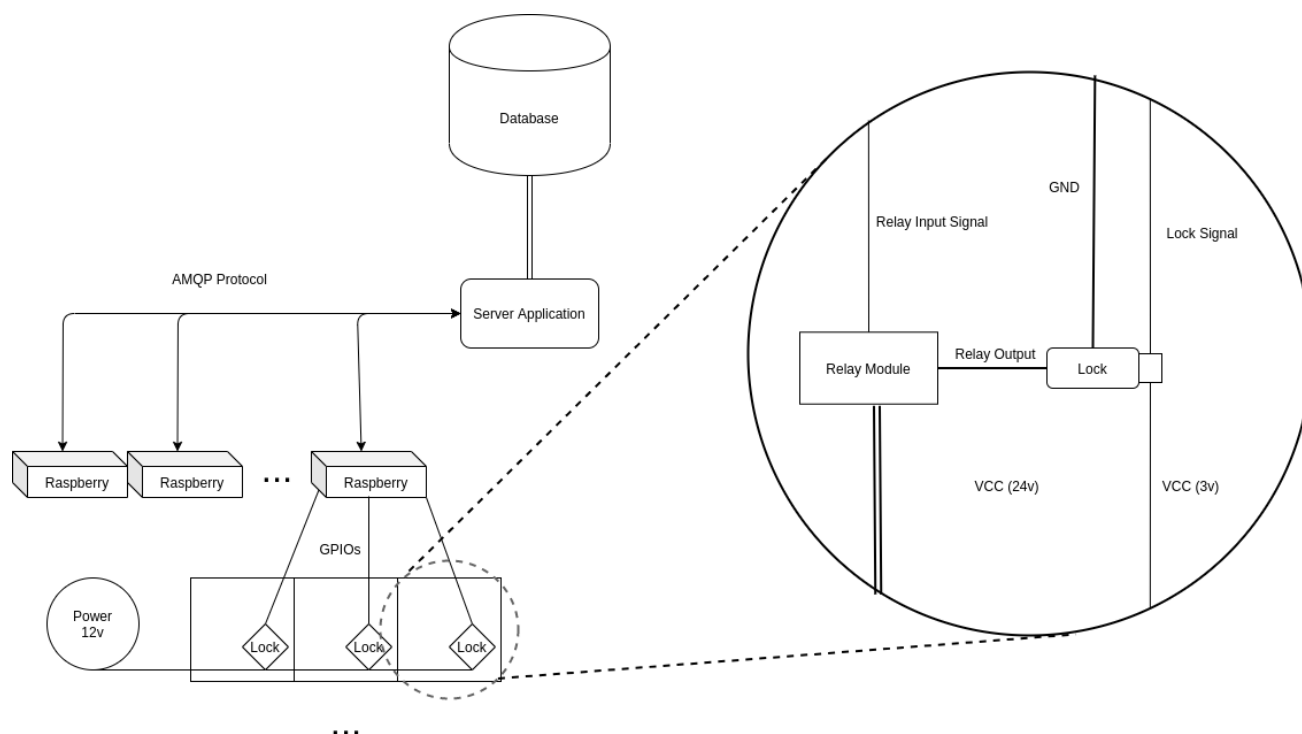
سمت کاربرسخت افزار

در ادامه به بررسی فنی بخش های مختلف می پردازیم.

۳.۱. معماری سخت افزار

در یک مکان ممکن است چندین مجموعه کمد وجود داشته باشد. در طراحی اولیه بخش سخت افزار، هر مجموعه کمد به وسیله یک برد رزبری پای کنترل می شود. این برد از طریق اینترنت و با استفاده از پروتوکل AMQP در یک ارتباط دو طرفه با سرور، دستور ها را می گیرد و اطلاعات لازم را به سمت سرور ارسال می کند. دلیل استفاده از این پروتوکل به جای REST این است که در این جا گاهی لازم است که اطلاعاتی از کمد ها به سرور منتقل شود پس باید ارتباط دوطرفه باشد. برد رزبری پای با استفاده یک اداپتور از برق شهری تغذیه می کند.

در شکل زیر یک شماتیک از معماری اولیه بخش سخت افزاری نمایش داده شده است.



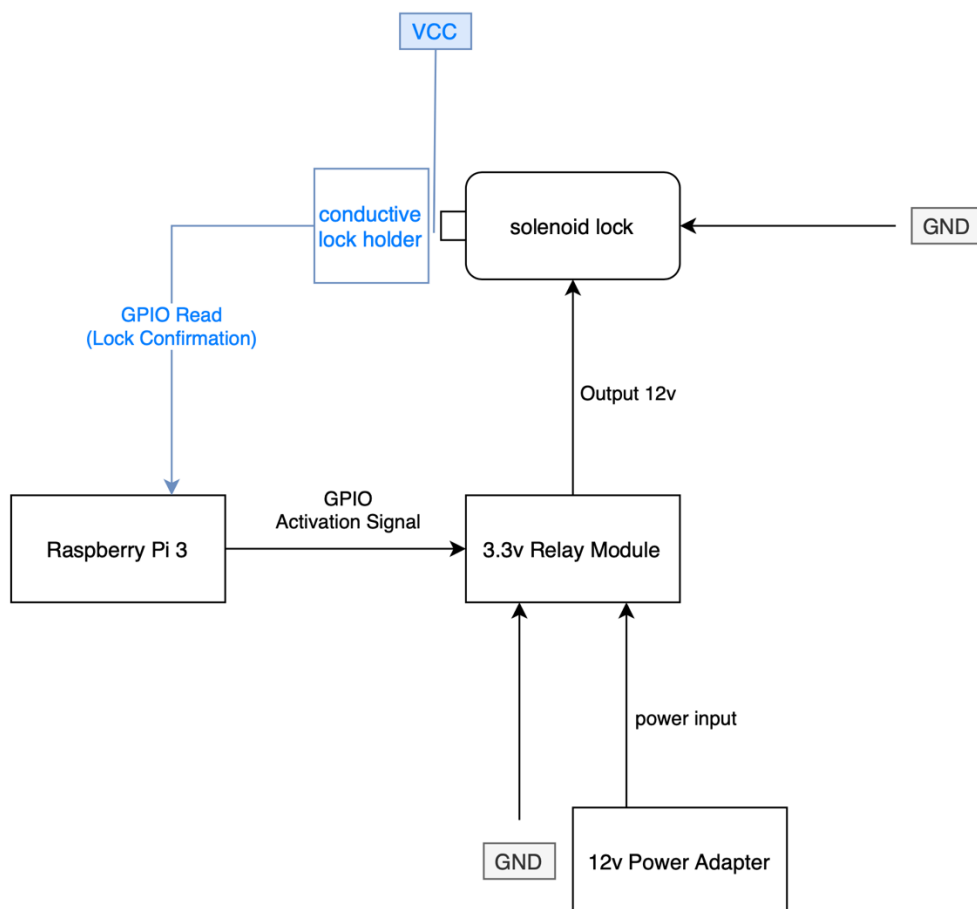
نمودار ارتباط سخت افزار با سرور و جزییات سخت افزاری

ابتدا برای هوشمند سازی باید قفل ها را با قفل های الکترونیکی تعویض کنیم. برای این کار از قفل های سلونویدی ۹ ولت - ۱.۵ آمپر استفاده می شود.

برای تامین ۹ ولت مورد نیاز این قفل ها باید از یک پاور خارجی استفاده کنیم چون رزبری پای توانایی تامین این ولتاژ و جریان را ندارد.

یک رله نیز برای اتصال پاور به قفل جهت تغذیه آن نیز وجود دارد. همانطور که در بخش بزرگنمایی شده تصویر مشاهده می شود با استفاده از GPIO رزبری پای (این بخش در طراحی های آینده ممکن است با ۴۸۵RS جایگزین شود)، سیگنال فعال سازی به رله ارسال شده و مدار وصل می شود و قفل باز می شود. برای بستن آن باید در را فشار داد تا قفل در محل خود گیر بیافتد.

برای تشخیص و مطمئن شدن از قفل شدن در، یک VCC بین زبانه قفل و محل گیر کردن آن وجود دارد که هنگام قفل بودن در، این VCC به مدار متصل می شود و در یکی از پایه های GPIO رزبری پای این سیگنال خوانده می شود. بخش آبی تصویر زیر مربوط به همین سیستم اطمینان از قفل شدن کمد است.



نمودار اتصالات سخت افزار در کمد با جزئیات بیشتر

لیست قطعات اصلی :

Raspberry Pi 3 B



Solenoid Lock
9v-12v/1.5A



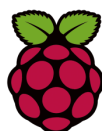
12v Relay Module



AC/DC 15W 12V
Power Supply



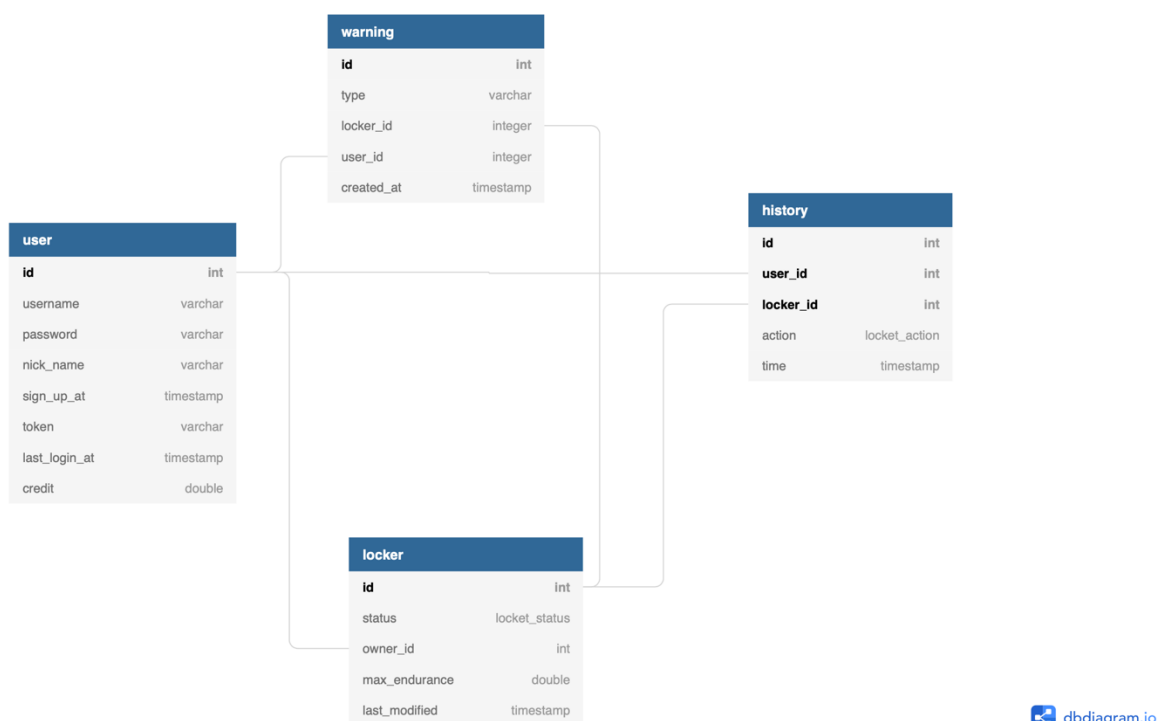
برای توسعه برنامه کنترل کننده کمد ها بر روی رزبری پای از زبان C++ به همراه کتابخانه RabbitMQ استفاده شده.



۳.۲. معماری سرور

برای معماری سرور ابتدا برای ارتباط کلاینت و سرور تحقیق و جستجو را آغاز کردیم. با توجه به کارایی و استفاده‌ی وسیع پروتکل REST به این نتیجه رسیدیم تا از این پروتکل برای ارتباط بین سرور و کلاینت استفاده کنیم. ویژگی بسیار خوب این پروتکل مستقل بودن از نوع platform و زبان برنامه‌نویسی بود که دست ما را در انتخاب پلتفرم‌ها و زبان‌های برنامه‌نویسی کاملاً آزاد می‌کرد. در ادامه به دلیل فراهم نبودن قطعات سخت‌افزاری و عدم طراحی معماری سخت‌افزار به طور کامل به سمت پیاده‌سازی بخش‌های مربوط به سرور حرکت کردیم.

در گام اول درباره ویژگی‌ها و امکانات سیستم دقیق شدیم و طراحی دیتابیس و جدول‌های مربوطه را آغاز کردیم.



شمای دیتابیس

با توجه به قابلیت‌های اولیه‌ای که برای کمد هوشمند در نظر گرفته بودیم، ۴ جدول فوق را طراحی و پیاده‌سازی کردیم. جدول user اطلاعات مربوط به هر کاربر را نگهداری می‌کند که شامل آیدی یکتا، نام کاربری، پسورد، اسم مستعار و تاریخ ثبت‌نام می‌باشد. جدول locker اطلاعات مربوط به هر کمد را نگهداری می‌کند که شامل شماره‌ی کمد، وضعیت فعلی کمد (باز/بسته)، آیدی صاحب فعلی کمد، میزان وزن آیتم‌هایی که هم‌اکنون در کمد قرار دارند، حداکثر میزان تحمل وزن و تاریخ و زمان آخرین تغییر ایجاد شده در کمد می‌باشد. یک جدول

دیگر نیز به نام **history** در نظر گرفتیم تا تاریخچه‌ی اکشن‌های اتفاق افتاده برای کمد‌ها را ذخیره‌سازی کنیم. این جدول شامل آیدی کاربر، شماره‌ی کمد، اکشن اتفاق افتاده (باز کردن/بستن/دریافت کمد/پس دادن کمد) و زمان روی‌دادن اکشن می‌باشد. در جدول **warning** نیز اخطار ها و هشدار هایی که باید به پنل ادمین برود نگهداری می شوند. این اخطار ها شامل موارد زیر می شود :

- باز شدن در کمد به صورت غیر مجاز و غیر از دستور صاحب کمد.
- کافی نبودن اعتبار کاربر برای گرفتن کمد در روز های آینده.
- بلا استفاده بودن یک کمد که در اختیار یک کاربر است برای یک مدت طولانی.

در گام دوم اقدام به تحقیق درباره‌ی نحوه‌ی پیاده‌سازی برنامه‌ی سرور کردیم و در نهایت به انتخاب فریم‌ورک **Spring boot** رسیدیم. از دلایل انتخاب این فریم‌ورک می‌توان به کارایی و سادگی در نحوه‌ی مدیریت **Endpoint** ها در پروتکل **REST** و پیاده‌سازی و ارتباط **entity** ها و دیتابیس اشاره کرد. در ادامه شروع به طراحی و پیاده‌سازی **API** های مورد نیاز کردیم. همچنین محاسبات هزینه روزانه کمد ها در سمت سرور پیاده سازی شده و اعتبار کاربران را کنترل می کند.

این **api** ها در سه صفحه پیش رو آمده اند.

Endpoints

- **/locker**

Endpoints for user interface:

- Headers
 - token: a string with length of 32 chars; use for authentication.
- **./status [GET]**
Get status of the user locker
- **./open [PUT]**
Open the user locker
- **./take [POST]**
Take a locker for user with input token
- **./return [PUT]**
Return the locker for user with input token

Endpoints for admin panel:

- **./info [GET]**
Get full information of all the lockers

- `/user`

Endpoints for user interface:

- `./sign_up` [POST]

Sign up a user into the system

- Request body
 - nickname
 - username
 - password
 - email
- Response body
 - status
 - token

- `./login` [POST]

Login user with given username and password

- Request body
 - username
 - password
- Response body
 - status
 - token

Endpoints for admin panel:

- `./search` [GET]

Get full information of the user with input username

- Request parameter
 - username

- `/locker/{id}`

Endpoints for admin panel:

- Path variable:
 - id: locker id to do an operation on.
- Headers
 - token: a string with length of 32 chars; use for authentication.

- `./status [GET]`

Get status of the locker for input id

- `./open [PUT]`

Open the locker for input id

- `./info [GET]`

Get full information of the locker with input id

- `/user/{id}`

Endpoints for admin panel:

- Path variable:
 - id: user id to do an operation on.

- `./info [GET]`

Get full information of the user with input id

- `/warning`

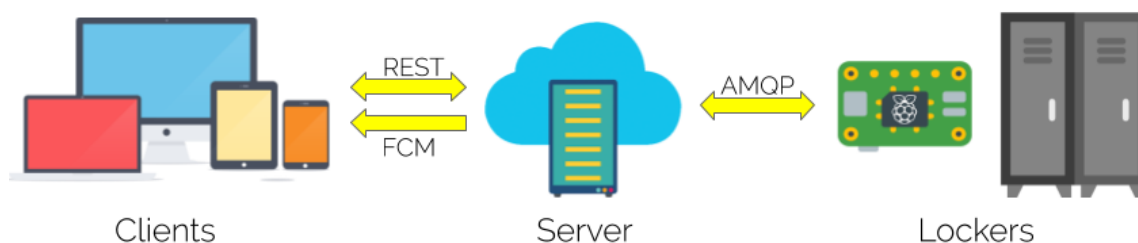
Get all warnings stored in the database. Use for Admin panel.

در گام سوم با همکاری تیم سخت‌افزار درباره‌ی پروتکل‌های ارتباطی بین سرور و سخت‌افزار تحقیق کردیم. با توجه به نیاز ارتباط دوطرفه (Bi-Directional) بین سرور و سخت‌افزار تصمیم بر آن شد تا از پروتکل AMQP استفاده کنیم. با توجه به انتخاب Spring Boot به عنوان فریم‌ورک مورد استفاده در برنامه‌ی سرور، تصمیم گرفتیم تا از پیاده‌سازی داخلی پروتکل AMQP این فریم‌ورک به نام RabbitMQ استفاده کنیم که کار پیکربندی و استفاده از این پروتکل را برای ما ساده کرده است.

از ویژگی‌های خوب دیگر AMQP علاوه بر ارتباط دوطرفه می‌توان به ماندگاری پیام‌ها بر روی صف اشاره کرد، که برخی موارد مانند زمانی که برنامه‌ی سرور و یا برنامه‌ی سخت‌افزار در دسترس نباشد می‌تواند به ما کمک کند تا اطلاعات مهم را از دست ندهیم. حالتی نیز وجود دارد که پیام‌های قدیمی برای ما مهم نباشند که می‌توانیم در هر بار اتصال به صف آخرین پیام را به عنوان پیام معتبر استفاده کنیم و سایر پیام‌ها را نادیده بگیریم و یا با تعریف ttl یا همان TimeToLive مدت زمان ماندگاری پیام در صف را مدیریت کنیم.

با توجه به ساختار قفل، پیغام تغییر وضعیت کمد به بسته از سمت سخت‌افزار برای سرور ارسال می‌شود و سرور باید به نحوی آن را به کاربر انتقال دهد. با توجه به این که برای ارتباط برنامه‌ی کلاینت و سرور از پروتکل REST استفاده کردیم و این پروتکل یک پروتکل یک‌طرفه می‌باشد، نمی‌توانیم از این طریق کاربر را مطلع سازیم. بنابراین شروع به جستجو درباره‌ی روش‌هایی برای ارسال پیام به سمت برنامه‌ی کلاینت کردیم. که در نهایت به پروتکل [FCM](#) یا همان [Firebase Cloud Messaging](#) رسیدیم که امکان ارسال پیام و نوتیفیکیشن را به سمت کلاینت فراهم می‌کند.

شمای کلی از معماری بخش سرور را می‌توانیم در تصویر زیر به طور خلاصه ببینیم.



۳.۳. رابط کاربری (مشتري و ادمين)

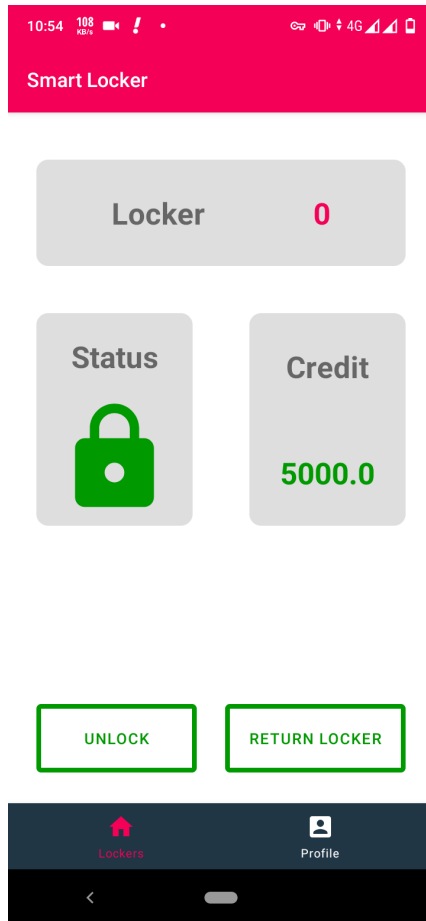
به منظور دسترسي افراد به كدها دو راه تعبيه شده است. روش اول استفاده از نرم افزار اندرويد اين سرويس مي باشد كه انتخاب اندرويد به عنوان پلتفرم اوليه به علت غالب بودن آن در سطح جامعه و به خصوص در ايران مي باشد. روش دوم براي دسترسي ادمين به كدها استفاده از نسخه وب برنامه است كه با توجه به وجود دستگاههاي جز اندرويد و يا دلايل شخصي افراد براي نصب نكردن نرم افزار، وجود آن ضروري به نظر مي رسد.

در سمت اندرويد برنامه با تكنولوژي هاي كاملا Native توسعه داده شده و زبان مورد استفاده نيز Kotlin مي باشد كه باعث بهبود عملکرد و گزينه هاي طراحي نسبت به Framework هاي هيبريدي خواهد شد. معماری برنامه نويسي در سمت كلاينت اندرويد معماری MVVM مي باشد كه معماری جديد و مورد تاييد شركت Google است.

در سمت ديگر يعني بخش وب از تكنولوژي هاي HTML/CSS به همراه Bootstrap براي طراحي رابط کاربري و AJAX براي ارتباط با سرور استفاده مي شود. از برتري هاي نسخه وب مي توان به وجود پنل مديريت براي حفاظت از اين سامانه اشاره كرد.



در ادامه تصاويري از رابط کاربري اندرويد و وب آمده و فايل هاي آن نيز پيوست شده است.



FUM Smart Locker

Admin Panel

☰ Menu

Overall Status

ID	Status	Owner
0	CLOSE	hojats7731
3	CLOSE	homz
6	CLOSE	dib
7	CLOSE	mvtikm
8	CLOSE	yavar

Warnings

ID	Message	Username	Email	Time
8	LOCKER_UNUSED	mvtikm	mvtikm@gmail.com	2020-07-15T21:41
0	UNEXPECTED_OPENING	mobinvahdati	mvtikm@gmail.com	2020-07-15T21:41
0	UNEXPECTED_OPENING	hojats7731	hojats7731@gmail.com	2020-07-15T21:41
7	NO_CREDIT	homz	p@gmail.com	2020-07-15T21:41