



## تمرین کامپیوتری شماره ۱



عنوان: مدیریت ورود و خروج بر بستر اینترنت اشیا

درس: سامانه های بی درنگ نهفته

اساتید: دکتر مهدی مدرسی – دکتر مهدی کارگهی

رشته: مهندسی کامپیوتر

طراح تمرین: سهراب مرادی

نیم سال دوم سال تحصیلی ۱۴۰۲-۰۳



عنوان پروژه.....	۱
۱-مقدمه.....	۱
سیستم‌های نهفته.....	۱
ابر (Cloud).....	۱
سیستم نظارتی.....	۱
۲-گام‌های انجام تمرین.....	۲
گام اول: نصب نرم‌افزار Proteus به همراه ماژول‌های مورد نظر.....	۲
گام دوم: پیاده‌سازی و اجرای وب سرور.....	۳
گام سوم: پیاده‌سازی نرم افزار نظارتی.....	۳
گام چهارم: گزارش نویسی و پاسخ به سوالات.....	۳
گام پنجم: بخش‌های امتیازی.....	۳
۳-جمع بندی و نکات پایانی.....	۴



## عنوان پروژه

مدیریت ورود و خروج بر بستر اینترنت اشیا

### ۱-تعریف مساله

در این تمرین قصد داریم تا سامانه‌ای جهت کنترل لحظه‌ای ورود و خروج افراد به یک شرکت فرضی را بر بستر اینترنت اشیا پیاده سازی کنیم. چنین سامانه‌ای متشکل از بخش‌های زیر خواهد بود:

- سامانه نهفته شامل سنسورها و ماژول‌های سخت افزاری
- ابر (Cloud) (در این تمرین یک وب سرور ساده در نظر گرفته می‌شود)
- سیستم نظارتی (نرم افزاری جهت مشاهده رویدادهای سامانه به شکل لحظه‌ای)

در ادامه به شرح هر بخش و ویژگی‌های مورد انتظار از آن می‌پردازیم.

### سامانه نهفته نصب شده در محل

جهت شناسایی کارمندان از سنسور و تگ RFID استفاده می‌شود. هر RFID tag برابر با هویت یک فرد است، همچنین ورودی شرکت یک RFID sensor قرار دارد که با استفاده از آن می‌توان مشخصات افراد را از روی RFID tag خواند. مشخصه هر فرد نیز یک کد ۱۰ رقمی است.

باز و بسته شدن در شرکت به وسیله یک servo motor کنترل می‌شود، همچنین مقابل در، یک چراغ LED و یک مانیتور وجود دارد. LED با توجه به اجازه ورود به دو رنگ قرمز یا سبز در می‌آید، در همین هنگام مشخصات فرد بر روی مانیتور نمایش داده می‌شود.

برای ارتباط با شبکه ی اینترنت و ارسال داده نیاز به یک سخت افزار جهت ارسال و دریافت داده به صورت بی سیم داریم که این کار با ماژول ESP8266 wifi انجام میشود.

در نهایت کنترل تمامی این ماژول‌ها به وسیله یک برد Arduino انجام می‌شود.

### ابر (Cloud)

متشکل از یک وب سرور که شامل لیستی از افراد مجاز است. تمامی افراد خارج از این لیست ورودشان به شرکت غیر مجاز است.

وب سرور و برد آردوینو از طریق پروتکل HTTP با یکدیگر ارتباط برقرار می‌کنند.

### سیستم نظارتی

این سیستم شامل یک نرم افزار است که از طریق WebSocket با وب سرور در ارتباط است و به محض ورود اطلاعات به وب سرور تمامی جزئیات بر روی این نرم افزار قابل مشاهده است.

## فلوچارت عملکرد سیستم

به صورت کلی تمام فرآیندی که به هنگام ورود هر شخص به شرکت انجام می شود به شرح زیر است:

- ۱) نزدیک شدن شخص (RFID tag) به ورودی شرکت.
- ۲) خواندن اطلاعات به وسیله RFID sensor.
- ۳) دریافت اطلاعات روی برد آردوینو.
- ۴) ارسال اطلاعات به ابر (اتصال به WiFi در فاز setup برد آردوینو انجام شده است).
- ۵) احراز هویت فرد در ابر
- ۶) ارسال اطلاعات هویت به سیستم نظارتی و نمایش تمامی جزئیات و ذخیره در تاریخچه.
- ۷) ارسال نتیجه احراز هویت به سمت برد آردوینو.
- ۸) در صورت تایید هویت انجام مراحل ۹ - ۱۲، در غیر این صورت انجام مراحل ۱۳ - ۱۵.
- ۹) سبز شدن چراغ LED.
- ۱۰) نمایش اطلاعات فرد بر روی نمایشگر (شامل شناسه ۱۰ رقمی و تاریخ و ساعت اقدام به ورود).
- ۱۱) باز شدن در (چرخش ۹۰ درجه ای servo motor).
- ۱۲) بسته شدن در پس از ۳۰ ثانیه.
- ۱۳) اگر در باز است در همان لحظه بسته شود.
- ۱۴) قرمز شدن چراغ LED.
- ۱۵) نمایش پیغام Access Denied.

**نکته:** فرض کنید در هر زمان فقط یک نفر برای احراز هویت به در نزدیک میشود.

## ۲- گام های انجام تمرین

**گام اول:** نصب نرم افزار **Proteus** به همراه ماژول های مورد نظر.

بخش نهفته تمرین بر روی نرم افزار پروتئوس شبیه سازی خواهد شد بنابراین لازم است تا این نرم افزار را به همراه ماژول های مورد نیاز بر روی سیستم خود نصب کنید. لازم به ذکر است با استفاده از این شبیه ساز و همچنین ماژول ESP8266 می توانید به یک WiFi واقعی متصل شده و درخواست های HTTP ارسال کنید (اتصال شبیه سازی شده است و فقط لازم است تا لپ تاپ شما به اینترنت دسترسی داشته باشد).

## گام دوم: پیاده‌سازی و اجرای وب سرور

لازم است تا یک وب سرویس ساده که از HTTP و WebSocket پشتیبانی می‌کند بر روی سیستم خود اجرا کنید. برای تجربه یکپارچه بر بستر ++C لازم است تا از [فریمورک Qt](#) برای ایجاد وب سرویس استفاده کنید. این فریمورک ابزار بسیاری برای ایجاد یک وب سرویس ساده در اختیار شما قرار خواهد داد و می‌توانید به سادگی وب سرویس مد نظر را پیاده‌سازی کنید.

## گام سوم: پیاده‌سازی نرم افزار نظارتی

برای پیاده‌سازی سیستم نظارتی نیز می‌توانید از فریمورک Qt استفاده کنید، قسمت GUI برنامه به صورت آماده در اختیارتان قرار خواهد گرفت و لازم است تا شما بخش‌های مربوط به هسته برنامه را در ++C پیاده‌سازی کنید که شامل یک وب سوکت (QWebSocket) برای برقراری ارتباط با سرور و توابع کمکی جهت ارتباط با رابط گرافیکی می‌شود.

## گام چهارم: گزارش نویسی و پاسخ به سوالات

دقت کنید گزارش نهایی شما می‌بایست همانند یک Document باشد و شامل توضیح کد و ساختار کد، همچنین نتیجه نهایی اجرای کد و اسکرین شات‌های دقیق از تمام مراحل باشد. (در فایل Readme.md کنار فایل اصلی خود و در Repo مربوطه قرار دهید.) **این نکته حائز اهمیت است که فایل PDF به هیچ عنوان مورد پذیرش قرار نخواهد گرفت.**

کامیت‌ها جزئی و دقیق باشند که برای این منظور می‌بایست به صورت ماژولار کد نویسی کنید و پس از اتمام هر ماژول، فرد نویسنده یک کامیت شامل ماژول پیاده‌سازی شده و ویژگی‌هایش انجام دهد، همچنین در صورت ایجاد تغییرات در ماژول‌هایی که قبلاً کامیت شده اند ذکر تغییرات لازم است.

در فایل Answers.md نیز به سوالات زیر پاسخ دهید:

- ۱) در بخش Embedded تمرین چگونه می‌توان بلوتوث را جایگزین RFID کرد؟ راه حل خود را شرح دهید.
- ۲) آیا می‌توان بدون نیاز به pair کردن ماژول‌های بلوتوث اطلاعات لازم (شناسه فرد) را بین آنها جابجا کرد؟ توضیح دهید.
- ۳) با فرض استفاده از بلوتوث چه راهکاری را برای جایگزینی استفاده از تایمر برای بسته شدن ورودی شرکت پیشنهاد می‌کنید؟ توضیح دهید.

## گام پنجم: بخش‌های امتیازی

- ۱) تاخیر انتها به انتها به صورت دقیق محاسبه و گزارش شود. (در اطلاعات ارسالی به سمت سرور لحظه‌ای دریافت اطلاعات نیز ارسال شود، در انتها، هم تاخیر نمایش گزارش بر روی نرم افزار کنترلی و هم تاخیر واکنش سیستم گزارش شود)
- ۲) ارتباط میان سیستم نهفته و وب سرور با استفاده از وب سوکت پیاده‌سازی شود.

### ۳- جمع بندی و نکات پایانی

- مهلت تحویل: در صفحه درس اعلام خواهد شد
- پروژه در گروههایی که قبلا تشکیل شده است انجام میشود. (گروه بندی در سامانه ایلرن نیز انجام می شود و تحویل تمرین **حضور**ی و به صورت **گروهی** خواهد بود)
- عضوهای گروه می بایست کار را تقسیم کنند. همچنین از Git برای ساختن branch و تقسیم issue ها استفاده نمایند. (با استفاده از commit ها و تعیین issue ها میزان مشارکت هر نفر مشخص می شود). بعد از انجام این کار کدها را در یک repository به نام CPS\_IOT\_HW در اکانت های GitHub خود قرار دهید (به صورت private). همچنین در یک فایل README.md می توانید report و داکيومنت خود را کامل کنید و در کنار repository قرار دهید. در نهایت لینک این repository را در محل پاسخ تمرین قرار دهید. (از فرستادن فایل به صورت زیپ جدا خودداری نمایید). اکانت تی ای های این تمرین رو به Repo خودتون به عنوان Maintainer به پروژه اضافه کنید.

#### Accounts:

@TheSohrab

- برای پیاده سازی این تمرین از ++C استفاده کنید.
- در انتها Repo شامل چهار فولدر اصلی خواهد بود: Server, Client, Embedded, Proteus.
- ساختار صحیح و تمیزی کد برنامه، بخشی از نمره این پروژه شما خواهد بود. بنابراین در طراحی ساختار برنامه دقت به خرج دهید.
- برای هر قسمت کد، گزارش دقیق و شفاف بنویسید. کدهای ضمیمه شده بدون گزارش مربوطه نمره ای نخواهند داشت.
- هدف این تمرین یادگیری شماست. لطفا تمرین را خودتان انجام دهید. در صورت مشاهده تشابه بین کدهای دو گروه، مطابق سیاست درس با گروه های خاطی برخورد خواهد شد.
- سؤالات خود را تا حد ممکن در **گروه درس** مطرح کنید تا سایر دانشجویان نیز از پاسخ آن بهره مند شوند. در صورتی که قصد مطرح کردن سؤال خاص تری دارید، از طریق ایمیل زیر ارتباط برقرار کنید.

[m.moradi1998@ut.ac.ir](mailto:m.moradi1998@ut.ac.ir) ○

موفق باشید.