



دانشگاه اصفهان

دانشکده مهندسي کامپیوتر

گروه مهندسی معماری کامپيوتر

گزارش پروژه کارشناسی

رشته‌ مهندسی کامپيوتر گرايش سخت‌افزار

عنوان پروژه:

پیاده‌سازی یک سامانه‌ی مدیریت آموزش

استاد راهنما:

جناب آقای دکتر جمشیدی

پژوهشگر:

مهدی جعفری

دی 1402

دانشگاه اصفهان

دانشکده مهندسي کامپیوتر

گروه مهندسی معماری کامپيوتر

پروژه کارشناسی رشته‌ي مهندسی کامپيوتر گرايش سخت‌افزار

آقای مهدی جعفری

تحت عنوان

پیاده‌سازی یک سامانه‌ی مدیریت آموزش

در تاريخ / / 13 توسط هيأت داوران زير بررسي و با نمره به تصويب نهايي رسيد.

1- استاد راهنماي پروژه:

دکتر کمال جمشیدی امضا

2- استاد داور :

دکتر امضا

امضاي مدير گروه

تشکر و قدرداني

بدین‌وسیله تشکر قلبی خود را از استاد عالی‌قدر جناب آقای دکتر کمال جمشیدی که در تمام مسیر، دلسوزانه راهنمایی این پایان‌نامه را عهده‌دار بودند و از هیچ کمکی دریغ ننمودند، ابراز می‌کنم و توفیقات روزافزون برای ایشان آرزو دارم.

همچنین از داوران گرامی که زحمت داوری این پایان‌نامه را عهده‌دار هستند کمال تشکر و قدردانی را دارم.

تقديم به

محضر گرامی خانواده‌ی مهربانم که همواره بزرگ‌ترین سرمایه‌ی من بوده و هستند و در تمام مسیر زندگی با فداکاری خود، خوب زیستن را به من آموخته‌اند.

چکيده:

پروژه تحت عنوان پیاده‌سازی یک سامانه‌ی مدیریت آموزشی انجام شده است. این سامانه با استفاده از زبان جاوا و فریم‌ورک Spring Boot، و در قالب دو برنامه‌ی متصل نوشته شده است. در این برنامه‌ها از ابزار Thymleaf برای ارتباط با بخش ظاهری (front-end) بهره گرفته شده است. همچنین بخش ظاهری آن با استفاده از زبان‌های HTML[[1]](#footnote-2)، CSS[[2]](#footnote-3)، JavaScript و فریم‌ورک Bootstrap و کتابخانه‌ی jQuery نوشته شده است. یکی از این دو برنامه بر روی سرور و برنامه دیگر بر روی سیستم‌های هر کلاس و یا آزمایشگاه اجرا می‌شود و برای مشاهده تمام عملکرد سامانه، اجرای هم‌زمان این دو الزامی است.

این سامانه به صورت کاملاً متن‌باز بر روی Github قرار داده شده است و امکان مشاهده تمام فرایند پیشرفت آن فراهم است. در این سامانه قابلیت ثبت‌نام و ورود دانشجویان، تعریف اساتید و کلاس‌های مختلف و همچنین قابلیت قراردادن اطلاعیه، امتحان و تکلیف در هر کلاس وجود دارد. علاوه بر ویژگی‌های مذکور، با استفاده از پردازش تصاویر دریافتی از هر کلاس به صورت بلادرنگ، امکان تشخیص حضور افراد در کلاس میسر است. این پردازش تصویر به کمک کتابخانه‌ی قدرتمند OpenCV[[3]](#footnote-4) که عموماً در سیستم‌های بینایی ماشین‌های بلادرنگ (real-time computer vision) کاربرد دارد، استفاده شده است.

واژگان کليدي: سامانه مدیریت آموزش، استفاده از پردازش تصویر در زبان جاوا، بینایی ماشین بلادرنگ، کتابخانه‌ی OpenCV در جاوا

[فصل اول مقدمه 7](#_Toc156560431)

[1-1- هدف پروژه 7](#_Toc156560432)

[1-2- کاربردهای پروژه 7](#_Toc156560433)

[1-3- ساختار پایان نامه 8](#_Toc156560434)

[فصل دوم مفاهیم اولیه 9](#_Toc156560435)

[2-1- مقدمه 9](#_Toc156560436)

[2-2- ایجاد پروژه 9](#_Toc156560437)

[2-3- ایجاد وابستگی‌های برنامه 9](#_Toc156560438)

[2-3- بهره‌گیری از Thymleaf 10](#_Toc156560439)

[2-4- دیتابیس 11](#_Toc156560440)

[2-4-1- MySQL 11](#_Toc156560441)

[2-4-2- XAMPP 11](#_Toc156560442)

[2-5- کنترل ورژن Git 12](#_Toc156560443)

[2-3-1- زیربخش سه سطحی اول 12](#_Toc156560444)

[2-2-2- زیربخش سه سطحی دوم 12](#_Toc156560445)

[شکل 2-1: قطع مسير A-B-D-F به علت تغيير مسير B 13](#_Toc156560446)

[2-؟- جمع‌بندي 14](#_Toc156560447)

[فصل سوم شرح پروژه 15](#_Toc156560448)

[3-1- مقدمه 15](#_Toc156560449)

[3-؟- جمع‌بندي 16](#_Toc156560450)

[فصل چهارم نتایج 17](#_Toc156560451)

[4-1- مقدمه 17](#_Toc156560452)

[4-؟- جمع‌بندي 18](#_Toc156560453)

[فصل پنجم نتیجه‌گیری و پیشنهادات 19](#_Toc156560454)

[پيوست 1: لیست برنامه‌ها 20](#_Toc156560455)

[منابع: 21](#_Toc156560456)

[شکل 2-4-1: نحوه‌ی کانفیگ دیتابیس در application.properties 11](#_Toc156560518)

[جدول 2-1: وابستگی‌های خارجی موردنیاز برای پروژه 9](#_Toc427459416)

مخفف‌ها:

|  |  |
| --- | --- |
| HTML  CSS  OpenCV  IDE | HyperText Markup Language  Cascading Style Sheets  Open Computer Vision  Integrated development environment |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# فصل اول مقدمه

## 1-1- هدف پروژه

هدف این پروژه، ارائه‌ی یک سامانه‌ی به‌روز برای مدیریت کلاس‌ها و آزمایشگاه‌ها است تا کاربران بتوانند در کمترین زمان، از امکاناتی مانند ثبت‌نام و ورود دانشجویان، تعریف اساتید و کلاس‌های مختلف و همچنین قابلیت قراردادن اطلاعیه، امتحان و تکلیف در هر کلاس، بهره‌مند گردند. علاوه بر اهداف مذکور، هدف اصلی و نوین این پروژه، تشخیص حضور افراد در هر کلاس و یا آزمایشگاه است که به صورت یک سیستم بینایی ماشین بلادرنگ پیاده‌سازی شده است. این سامانه همچنین امکان ذخیره‌ی تصویر افراد حاضر در کلاس را نیز میسر می‌کند. خروجی این سامانه بر روی مرورگر کاربران قابل‌استفاده است.

## 1-2- کاربردهای پروژه

در عصری زندگی می‌کنیم که بسیاری از رفتارهای تکراری و خسته‌کننده، توسط هوش مصنوعی و ابزارهای جدید جایگزین می‌شوند و باارزش نهادن به زمان افراد، موج عظیمی از فعالیت‌ها به کامپیوترها سپرده خواهند شد. یکی از اصلی‌ترین نیازهای موجود در مجموعه کلاس‌های هر دانشگاه، فقدان اطلاعات درباره پر یا خالی بودن کلاس‌های آن مجموعه است. این اطلاعات معمولاً با پرسش‌وپاسخ از مسئولین مرتبط و یا سرکشی حضوری به محل موردنظر حاصل می‌شوند. درحالی‌که چنین رفتاری در عصر اطلاعات پذیرفتنی نیست. این سامانه علاوه بر ارائه خدمات سامانه‌های سابق، این تقاضا را نیز به‌راحتی برآورده می‌کند.

باتوجه‌به موارد گفته شده، می‌توان کاربرد بسیاری برای بخش پردازش تصویر این سامانه در مجموعه‌های آموزشی، سالن‌های ورزشی و حتی مجموعه‌های تفریحی و مراکز اداری، متصور بود. کاربر موردنظر تنها با کسب دسترسی لازم و چند کلیک، می‌تواند از وضعیت محیط موردنظر آگاه شده و نسبت به مدیریت آن اقدام کند. لازم به ذکر است که امکان استفاده از هر بخش این سامانه به صورت جعبه سیاه (Black Box) نیز وجود دارد. برای مثال می‌توان صرفاً از بخش پردازش تصویر آن در واحد‌های دیگر بهره گرفت و خروجی RestTemplate آن را دریافت کرده و در اپلیکیشن‌های دیگر نیز مورداستفاده قرارداد.

## 1-3- ساختار پایان نامه

ابتدا در فصل دوم، به بررسی فرایند ایجاد پروژه و ایجاد اتصالات دیتابیس مربوطه می‌پردازیم. چالش‌ها و نکات مربوط به تنظیمات اولیه چنین پروژه‌ای را مرور خواهیم کرد. همچنین فرایند اتصال پروژه به گیت‌هاب از مواردی است که در فصل دوم به آن پرداخته خواهد شد.

پس از بیان مفاهیم راه‌اندازی اولیه، در فصل سوم به سراغ فرایند کدنویسی، تعریف کلاس‌ها و روابط آن‌ها، کامپوننت نویسی و طراحی قالب برنامه، ایجاد صفحات مربوطه، نمایش پیغام‌ها، شیوه بررسی دسترسی‌ها و مدیریت فرایند امنیت سامانه خواهیم رفت و چالش‌های این مسیر مورد بررسی قرار خواهند گرفت.

در نهایت در فصل چهارم، مباحث مربوط به پردازش تصویر، شیوه‌های پیاده‌سازی، چالش‌ها و راه‌حل‌های آن‌ها مطرح خواهند شد.

# فصل دوم مفاهیم اولیه

## 2-1- مقدمه

در این فصل، ابتدا به شیوه ایجاد پروژه، تعریف وابستگی‌ها و ایجاد اتصالات دیتابیس مربوطه پرداخته خواهد شد. چالش‌ها و نکات مربوط به تنظیمات اولیه چنین پروژه‌ای را مرور خواهند شد. همچنین مبحث کنترل ورژن از مواردی است که در فصل مورد بررسی قرار می‌گیرد.

## 2-2- ایجاد پروژه

زبان جاوا به همراه فریم‌ورک Spring Boot یک بستر بسیار مناسب برای توسعه برنامه‌های سازمانی و سیستم‌هایی است که قرار است در مقیاس بزرگ و به صورت کاملاً ایمن و پایدار توسعه داده شوند. برای ایجاد پروژه می‌توان به سایت start.spring.io مراجعه نمود. با پر کردن فیلدهای مربوطه، قادر به دریافت پروژه‌ی خام خود خواهیم بود. برای توسعه و کدنویسی پروژه‌های Spring Boot از IDE[[4]](#footnote-5) های مختلفی می‌توان استفاده کرد که معروف‌ترین آن‌ها InteliJ محصول شرکت JetBrains است.

## 2-3- ایجاد وابستگی‌های برنامه

مفهوم وابستگی (Dependency) از مفاهیم پایه در زبان‌های شیءگرا است که از بازنویسی کدها جلوگیری می‌کند. این وابستگی‌ها می‌توانند داخلی و یا خارجی باشند که در این قسمت وابستگی‌های خارجی موردنیاز معرفی شده‌اند. این وابستگی‌ها در فایل pom.xml قابل‌تعریف هستند.

جدول 2-1: وابستگی‌های خارجی موردنیاز برای پروژه

|  |  |
| --- | --- |
| artifactId | GroupId |
| spring-boot-starter-web | org.springframework.boot |
| spring-boot-starter-actuator | org.springframework.boot |
| spring-data-jpa | org.springframework.data |
| spring-boot-starter-data-jpa | org.springframework.boot |
| mysql-connector-java | mysql |
| ojdbc10 | com.oracle.database.jdbc |
| spring-boot-starter-thymeleaf | org.springframework.boot |
| bootstrap | org.webjars |
| jquery | org.webjars.bower |
| spring-boot-starter-tomcat | org.springframework.boot |
| opencv | org.openpnp |

## 2-3- بهره‌گیری از Thymleaf

Thymeleaf یک ابزار مفید برای استفاده از زبان HTML در پروژه‌های Java است که به شما این امکان را می‌دهد که قالب‌های HTML را به صورت پویا در برنامه‌های Spring Framework استفاده کنید. از Thymeleaf معمولاً در تولید و نمایش داده‌ها در صفحات وب استفاده می‌شود.

همان‌طور که در بخش قبل نیز اشاره شد، تعریف وابستگی آن تحت عنوان spring-boot-starter-thymeleaf به فایل pom الزامی است. همچنین برای مهیاشدن شرایط طراحی قالب پروژه، در پوشه‌ی resources دو پوشه‌ی جدید با عنوان‌های static و templates ایجاد شده است.

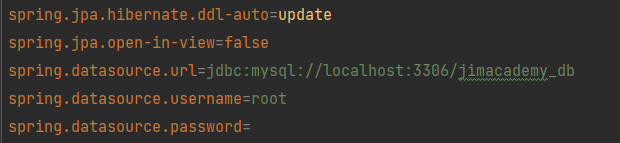
## 2-4- دیتابیس

بسیاری از برنامه‌های نرم‌افزاری نیاز به ذخیره‌سازی و مدیریت داده دارند و دیتابیس‌ها به عنوان یک ذخیره‌ساز مرکزی و مؤثر برای این منظور استفاده می‌شوند. در موارد زیادی، دیتابیس به عنوان یک منبع اطلاعات مهم برای برنامه‌ها عمل می‌کند و امکانات مختلفی را برای مدیریت و جستجو در داده‌ها ارائه می‌دهد. در این پروژه نیز برای ذخیره‌سازی داده‌های مربوط به کلاس‌ها، دانشجویان، تکالیف و تمرینات، اطلاعات کاربران و هم‌چنین روابط میان این موارد به یک دیتابیس مستقر بر روی سرور احتیاج دارد.

### 2-4-1- MySQL

نرم افزار MySQL، یک سیستم مدیریت دیتابیس است که انتخاب خوبی برای توسعه وب‌سایت‌ها یا برنامه‌های کاربردی وب به حساب می‌آید. MySQL نرم افزاری منبع‌باز ست که توسط Oracle پشتیبانی می‌شود.

باتوجه‌به راحتی و مدیریت آسان، در این پروژه نیز از این سیستم مدیریت دیتابیس استفاده شده است که وابستگی آن نیز قبلا ذکر شده است. برای اتصال پروژه به دیتابیس مذکور، در فایل application.properties نیاز به معرفی آن داریم که به صورت زیر انجام گرفته است:



شکل 2-4-1: نحوه‌ی کانفیگ دیتابیس در application.properties

لازم به ذکر است که باتوجه به استقرار دیتابیس بر روی لوکال، رمزعبور تعریف نشده است. هم‌چنین در عبارت نخستین، مدیریت جداول دیتابیس، مانند ایجاد، حذف و تغییر نیز به فریم‌ورک سپرده شده است. این مدیریت بر حسب انوتیشین‌های تعریف شده در پروژه صورت می‌گیرد که در فصل بعد به آن‌ها می‌پردازیم.

### 2-4-2- XAMPP

## 2-5- کنترل ورژن Git

Git یک سیستم کنترل نسخه توزیع شده است که برای مدیریت کد منبع و تاریخچه تغییرات در پروژه‌های نرم‌افزاری استفاده می‌شود. در Git، هر پروژه دارای یک مخزن (repository) محلی است که حاوی کلیه تاریخچه تغییرات و شاخه‌های مختلف کد می‌باشد. توسعه‌دهندگان می‌توانند تغییرات خود را اعمال کنند، شاخه‌های جدید ایجاد کنند، با شاخه‌های دیگر ادغام کنند، و تغییرات را به مخزن اصلی master منتقل کنند و پیشرفت پروژه را مشاهده کنند.

در این پروژه، از پلتفرم Github استفاده شده است. همان‌طور که پیش‌تر اشاره شد، این سامانه به صورت دو پروژه‌ی مرتبط پیاده سازی شده است و در پلتفرم GitHub نیز این دو پروژه به صورت مجزا آپلود شده‌اند که لینک آن‌ها در پیوست ضمیمه شده است.

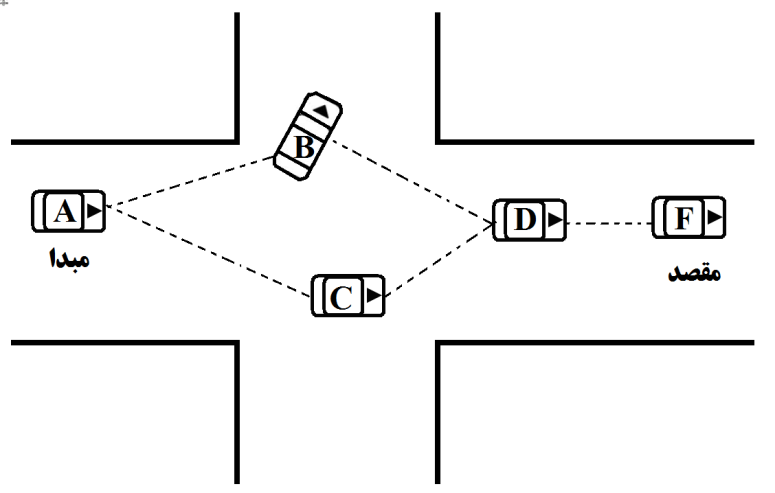
### 2-3-1- زیربخش سه سطحی اول

اگر باز هم زیر بخش نیاز داشتید می‌توانید شماره گذاری را به شیوه فوق انجام دهید. (فونت آن B nazanin و به صورت Bold و با اندازه 14 است. توجه داشته باشید که اگر در عنوان کلمه انگلیسی وجود داشت، باید فونت آن را 12 قرار دهید).

### 2-2-2- زیربخش سه سطحی دوم

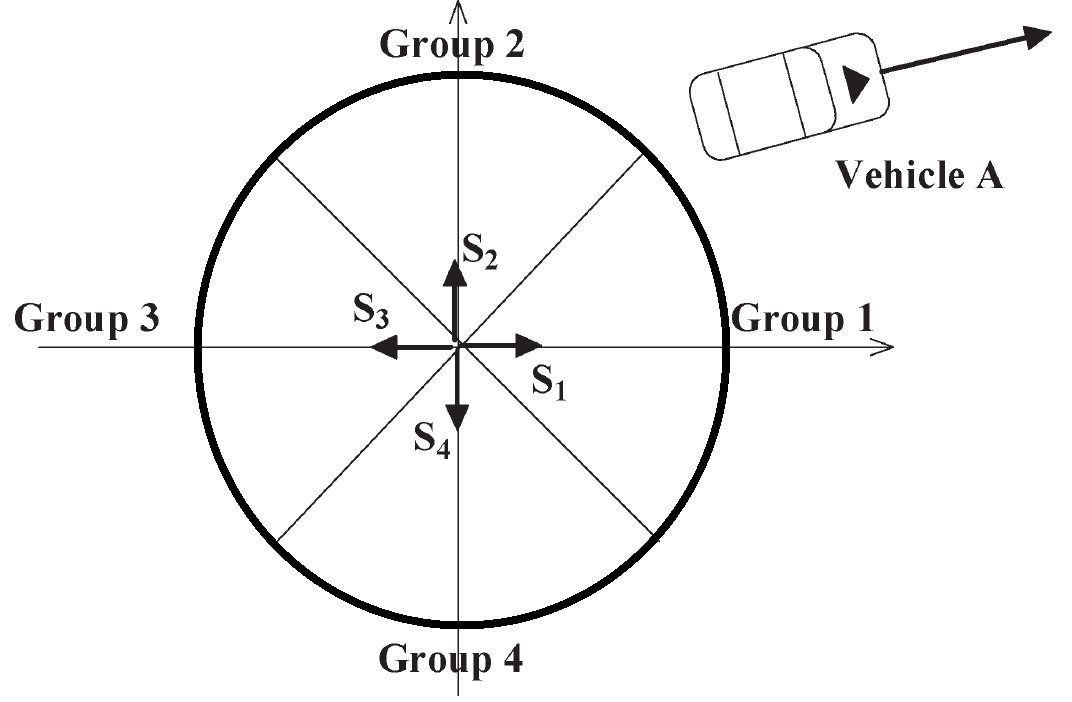
به نحوه شماره گذاری‌ها دقت کنید. اگر در متن کلمه مخفف داشتید، باید زیرنویس بدهید و در قسمت مخففها نیز آن کلمه را بیاورید. مثلا برای کلمه FPGA[[5]](#footnote-6) زیر نویس داده شده است[3].

#### 2-2-2-1- مثالی از آوردن شکل



شکل 2-1: قطع مسير A-B-D-F به علت تغيير مسير B

هر شکل حتما باید عنوان داشته باشد و همچنین شماره آن نیز مشخص شده باشد. عنوان شکل باید وسط چین باشد و فونت آنBold 12 B Nazanin باشد. اگر کلمه انگیسی در عنوان وجود داشت فونت آن باید Times New Roman Bold 11 باشد.



شکل 2-2: گروه‌بندي گره‌ها بر اساس بردار سرعت

#### 2-2-2-1- مثالی از آوردن جدول

جدول نیز مانند شکل باید شماره و عنوان داشته باشد با این تفاوت که عنوان جدول باید بالای جدول گذاشته شود. در صفحه بعد یک جدول نمونه آورده شده است.

جدول 2-1: جهت‌هاي انتخابي اتومبيل‌ها و زاويه بين راستاهاي حرکت آن‌ها

|  |  |  |
| --- | --- | --- |
| زاويه بين راستاي حرکت تومبيل‌ها | جهت انتخابي اتومبيل F | جهت انتخابي اتومبيلB |
| صفر درجه | چپ | چپ |
| صفر درجه | راست | راست |
| صفر درجه | مستقيم | مستقيم |
| 90 درجه | چپ | مستقيم |
| 90 درجه | راست | مستقيم |
| 90 درجه | مستقيم | چپ |
| 90 درجه | مستقيم | راست |
| 180 درجه | راست | چپ |
| 180 درجه | چپ | راست |

## 2-؟- جمع‌بندي

همانطور که گفته شد، در انتهای هر فصل یک زیر بخش به نام جمع‌بندی وجود دارد که خلاصه فصل و نتیجه فصل را در یک پاراگراف توضیح می‌دهید.

# فصل سوم شرح پروژه

## 3-1- مقدمه

این فصل نیز همانند بقیه فصلها با مقدمه شروع و نهایتا با جمع‌بندی ختم می‌شود.

## 3-؟- جمع‌بندي

# فصل چهارم نتایج

## 4-1- مقدمه

## 4-؟- جمع‌بندي

# فصل پنجم نتیجه‌گیری و پیشنهادات

در اين فصل نتیجه‌گیری و پیشنهادات را ذکر می‌کنید(کمتر از 5 صفحه باشد)

## پيوست 1: لیست برنامه‌ها

در قسمت پیوست می‌توانید لیست برنامه‌های نوشته شده را بیاورید. البته برنامه‌ها و ضمایم آن را به صورت CD باید به گزارش نهایی ضمیمه کنید.

## منابع:

[1] L. Andreone, and C. Ricerche, “Activities and applications of the vehicle to vehicle and vehicle to infrastructure communication to enhance road safety,” in *5th European Congress and Exhibition of Intelligent Transportation System*, Hannover, Germany, Jun. 2005.

[2] R. Morris, J. Jannotti, F. Kaashoek, J. Li, and D. Decouto, “CarNet: a scalable ad hoc wireless network system,” in *ACM SIGOPS European Workshop, beyond the PC: New Challenges for the Operating System*, Kolding, Denmark, 2000, pp. 61–65.

[3] T. Taleb, E. Sakhaee, A. Jamalipour, K. Hashimoto, N. Kato and Y. Nemoto, “A stable routing protocol to support ITS services in VANET networks.” *IEEE Trans. Veh. Technol.*, vol. 56, no. 6, pp. 3337 – 3347, 2007.

1. HyperText Markup Language [↑](#footnote-ref-2)
2. Cascading Style Sheets [↑](#footnote-ref-3)
3. Open computer vision [↑](#footnote-ref-4)
4. Integrated Development Environment [↑](#footnote-ref-5)
5. Field Programmable Gate Array [↑](#footnote-ref-6)