1. Online Exam portal

Teacher Requirement

- Teacher can schedule exam (subject wise)
- Teachers can add MCQ questions with images and text.
- Students will participate in the exams and according to exam result analytics will show that in which subject the student is good or not good.
- Teachers can check performance of students

Student Requirement

- Students can attend exams at specified time.
- Automatically the exam will be submitted when time is up.
- If the app is minimised then the user exam gets submitted or out of the exam.
- After submit the exam, automatically result will be displayed regarding performance
- Analytics in which subject student is good and in which is not good.

What to Submit

- ☐ Use only Kotlin for back-end and front-end
- ☐ Provide a URL to your working prototype.
- ☐ Provide a URL to your GitHub code repository. Repository must be public
- ☐ Include a video (3-5 minutes) that demonstrates your project in action (hosted on YouTube, LinkedIn, or Facebook Video and made public). Video must include an explanation.

2. School Management

Teacher -

- Teachers can create homework and assign it to a particular class .
- Teachers can filter students' homework is done or not done.
- Can send a warning through notification to a particular student through an app.
- Teacher can update performance of student
- Inform holidays, notice, information, PTM, Timetable.
- Syllabus Track
- Everyday’s attendance information of students and for particular subjects also.

- Late coming time for students.
- Fees Submit
- Approve leave application
- Fee detail
- Book Assign by Library
- Upload Notes

Student -

- Parents/Students can select when their homework is done.
- Parents can check the leaves of their child.
- Can get the status of how many times he is being late for school.
- Performance in particular subject
- Can see warning of students
- Can get information about notice,PTM, Holidays, Timetable.
- Submit leave application form
- Fee details
- Track Syllabus
- Book taken from library
- Students Documents
- Notes for Students

What to Submit

- ☐ Use only Kotlin for back-end and front-end
- ☐ Provide a URL to your working prototype.
- ☐ Provide a URL to your GitHub code repository. Repository must be public
- ☐ Include a video (3-5 minutes) that demonstrates your project in action (hosted on YouTube, LinkedIn, or Facebook Video and made public). Video must include an explanation.

3. EV Vehicles

App for Ev vehicles add when was the last charge was done, how much km it has to be run , on map show where are the charging points are there to charge your vehicles,

- Update then check last charge from stations and ride kilometres
- Show Nearby Charging Stations different legend for different things
    - Home Charging Station
    - Power Charging Stations
    - Ev repair

- Route of Charging station through map
- Comment , rating , report of particular ev station
- Add a new charging station
    - From Home
    - Charging station
    - Ev Repair Shops
- Payment integration - recharge the battery and make the payment
- Buy any parts of EV vehicles listed in the application.
- History from where last time got charged
- Battery Health Status.

What to Submit

- ☐ Use only Kotlin for back-end and front-end
- ☐ Provide a URL to your working prototype.
- ☐ Provide a URL to your GitHub code repository. Repository must be public
- ☐ Include a video (3-5 minutes) that demonstrates your project in action (hosted on YouTube, LinkedIn, or Facebook Video and made public). Video must include an explanation.

4. Car Service

Develop an app which shows nearby service centres for cars and book for car services

Workshop Requirement

- Service centre can add their workshop,Multiple services and their price with deals on app
- Can see bookings with notification
- Bookings history
- Update Status regarding car service

User Requirement

- Nearby service centres
- Can see full detail regarding services, prices , deals.
- User can book slot for their car service
- Option if the user wants doorstep pickup/delivery before and after service.
- Check the status of car service
- Rate and comment after service.
- Payment Option

- Offline Payment
- Online payment Through wallet and credit/debit card, UPI

What to Submit

- ☐ Use only Kotlin for back-end and front-end
- ☐ Provide a URL to your working prototype.
- ☐ Provide a URL to your GitHub code repository. Repository must be public
- ☐ Include a video (3-5 minutes) that demonstrates your project in action (hosted on YouTube, LinkedIn, or Facebook Video and made public). Video must include an explanation.

5. **Ecommerce with AR/VR:**

   Create Ecommerce apps which use AR or VR technology to show products to clients. Like-: Try products with AR, can see products in 360 degree with all information and features to point to that specific point.

What to Submit

- ☐ Use only Kotlin for back-end and front-end
- ☐ Provide a URL to your working prototype.
- ☐ Provide a URL to your GitHub code repository. Repository must be public
- ☐ Include a video (3-5 minutes) that demonstrates your project in action (hosted on YouTube, LinkedIn, or Facebook Video and made public). Video must include an explanation.

6. **Communication between doctors and patients:**
   An All in One system which makes technical things understandable to the patients, like patients can check their reports and also understand what is wrong or what is right in their report and also the same system will be helpful for the doctors so they can communicate with patients in a better way. Ex-: Appointment Booking, Colour scheme for patients disease, doctor availability, Medical Billing etc.

What to Submit

- ☐ Use only Kotlin for back-end and front-end
- ☐ Provide a URL to your working prototype.
- ☐ Provide a URL to your GitHub code repository. Repository must be public
- ☐ Include a video (3-5 minutes) that demonstrates your project in action (hosted on YouTube, LinkedIn, or Facebook Video and made public). Video must include an explanation.

7. **Ambulance management:**
   It is the 21st century and we are still managing ambulances on phone calls, we can automate it with gps and Ola/Uber like systems.

Multi-Platform

8. **Task management**
   - Can Make Group and Add members from contact list.
   - Can Create Task and assign to members who are on App from his contact
   - Users can update status of task
   - Status of Task to be updated on WhatsApp also
   - Task Reminder should come in floating notification like messenger.
   - User can create task and reminder for himself/herself also option like once or daily
   - Setting for particular task with sound or without sound notification
   - Delete Task.
   - History of task and from history user again assign task for future
   - Share particular task

What to Submit

- ☐ Use only Kotlin for back-end and Kotlin Multiplatform for front-end
- ☐ Provide a URL to your working prototype.
- ☐ Provide a URL to your GitHub code repository. Repository must be public
- ☐ Include a video (3-5 minutes) that demonstrates your project in action (hosted on YouTube, LinkedIn, or Facebook Video and made public). Video must include an explanation.

9. Fleet management

- vehicles added in the db by the fleet manager.
- Check Ins logs to be maintained (added by the drivers) when picking up the vehicle.
- After checking in the vehicle, the driver will fill mandatory OKAY checks. (eg: Side mirror is OKAY, Tires are OKAY)
- Driver Can update the status ex:- Stopped vehicle for dinner.
- Fuel Information by driver. (Upload Fuel receipts)
- On Dropping of the vehicle driver will fill up the mandatory OKAY checks and logs will be maintained in the DB on Dropping off with the time stamps.
- Fleet managers will be able to see all the stats and information about the vehicles and track vehicles through the driver's GPS.

What to Submit

- ☐ Use only Kotlin for back-end and Kotlin Multiplatform for front-end
- ☐ Provide a URL to your working prototype.
- ☐ Provide a URL to your GitHub code repository. Repository must be public
- ☐ Include a video (3-5 minutes) that demonstrates your project in action (hosted on YouTube, LinkedIn, or Facebook Video and made public). Video must include an explanation.

10. DOOH and OOH advertising management application and software

- OOH is one of the most evergreen modes of advertising. DOOH is its digital version which has seen immense growth and acceptance in the past few years.
- What if you can develop a mobile application that can act as an eCommerce portal and a platform between the customers and the OOH advertising company

- With a mobile application in place, the OOH companies can do various tasks seamlessly like campaign planning, reports and analytics, secure financial securities, easy management, and real-time modification

What to Submit

- [ ] Use only Kotlin for back-end and Kotlin Multiplatform for front-end
- [ ] Provide a URL to your working prototype.
- [ ] Provide a URL to your GitHub code repository. Repository must be public
- [ ] Include a video (3-5 minutes) that demonstrates your project in action (hosted on YouTube, LinkedIn, or Facebook Video and made public). Video must include an explanation.