Team 2

Golden Pegasus

Helping you stay productive, using your mobile context

Rachel Gan, Young Woong Jun, Sigrid Marita Kvamme, Jin Yeong Kim, Seung Woo Jo

Project Deadline: Dec. 9 Midnight

• Make sure all the project related materials and pushed to the git repo.

• Source code, design images, documents, presentation slides.

• Do not commit / push anything to the git repo after the

project deadline.

• Push your final presentation slides before the deadline. Upload it to etl as well.

Final Presentation

• Demonstration of your application (Must!)

• Suggest a couple of common use case scenarios and demonstrate the applications for those scenarios.

- Project overview
  - Motivation
  - Proposed idea
  - Novelty
- Technical details
  - Libraries and tools used
  - System architecture overview
  - Technical challenges faced
  - Solution approaches
  - Evaluation results (if any)
- Project management

• Scope of the project

• What were added and removed from the original plan and revised plan during the midterm presentation.• Timeline

• What was done and when (e.g., weekly breakdown will be nice)• Roles and contributions

• Who did what• Lessons learnt

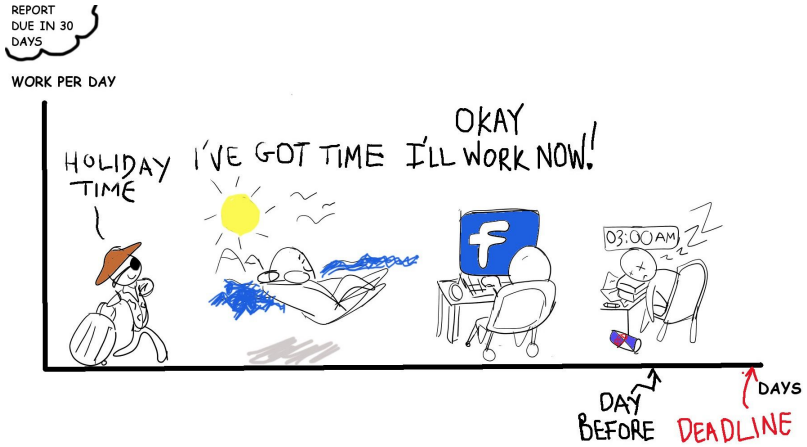
Project Idea Interactive Productivity Manager

Target Audience

- Students or workers who have problems meeting deadlines
- Procrastinators and people who get distracted easily

Goal

- Prevent users from procrastinating and help them to increase their productivity while staying happy



Final Deliverable & Success Criteria

Todo List & Homework Mode

- Stable homework mode that keeps track of time while the mode is on.
- Send notification and fire the alarm at proper time.
- Accurate detection of user motions (shaking and walking)

App Blocking

- Successfully block selected apps based on settings configuration
- Unblock app after step count completed and/or timer completed

Location-Reminder

- Stable location tracking and travel time estimation
- Send notification and fire the alarm at proper time.
- Reward and penalize properly in gamification

DEMO

Libraries and Tools used

AppBlocking

- UsageStatsManager
- GoogleFit Api

Location Reminder

- Google GPS API
- Naver Searchplace API
- ODSay API
- Daimajia SwipeLayout

Todo

- Android Room Database
- Material Progressbar for timer

Feature

Homework mode & To-do list

From the last presentation To-Do List & Homework Mode

Things need to be done

- Stable Timer
- Send Notification 3 minutes prior to timer ends with vibration (4 seconds after timer starts in Demo version)
- Make alarm go off as timer ends
- Use motion data to turn off the alarm

Expected technical challenges

- Keeping the timer going as the app goes to the background
- Detecting user motions (especially walking)
- Making sensors alive only when they are needed

To-Do List & Homework Mode Challenges

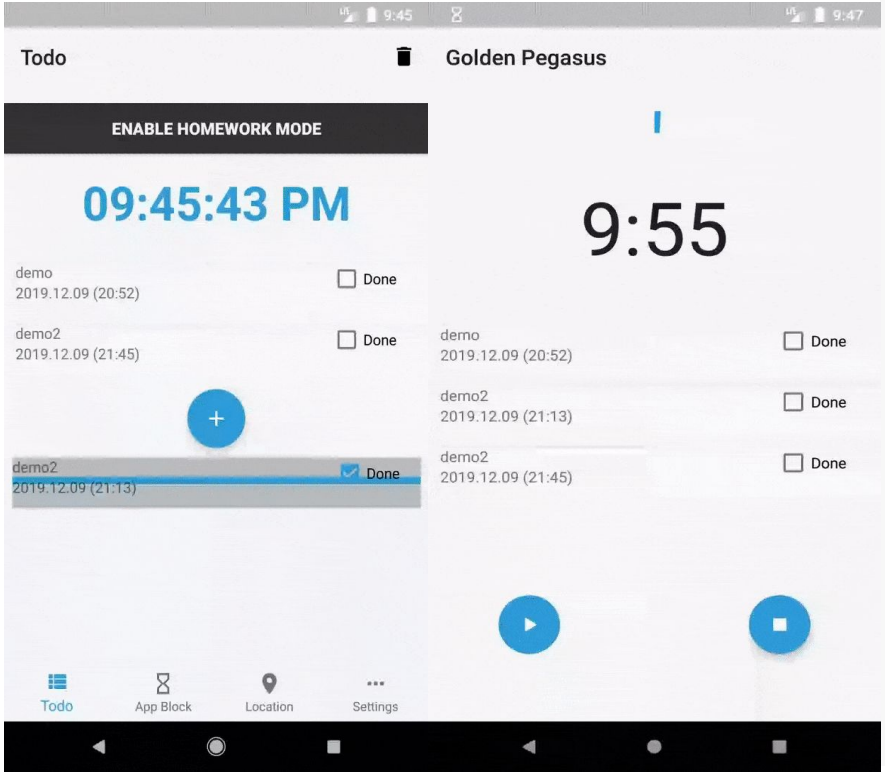
Challenges

- Working with time in android was very cumbersome task(Timer, Alarm)
- Controlling the sensor object (making it alive only when it is needed) was also cumbersome
- Notification has been quite easy compared to other parts

How were the challenges handled?

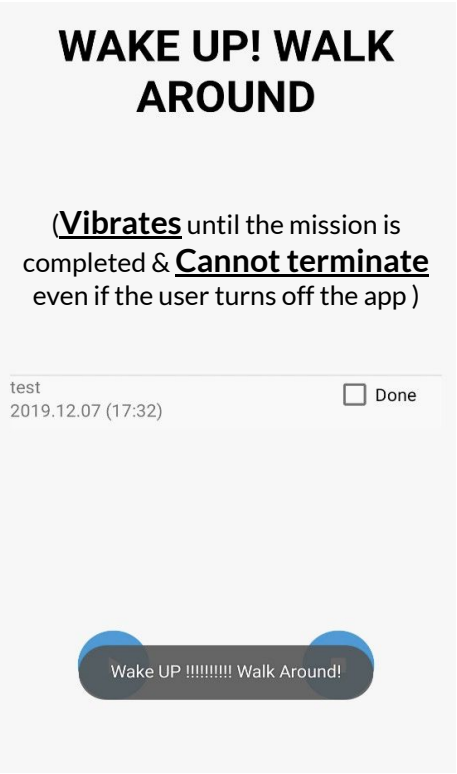
- Stable Timer & Alarm : Accomplished by using shared preferences and properly catching android activity lifecycle
- Sensor : It’s only alive when it’s needed (shaking, walking mission) by registering / unregistering at appropriate timing & Checking if the sensor object is already instantiated

To-Do List & Homework Mode DEMO

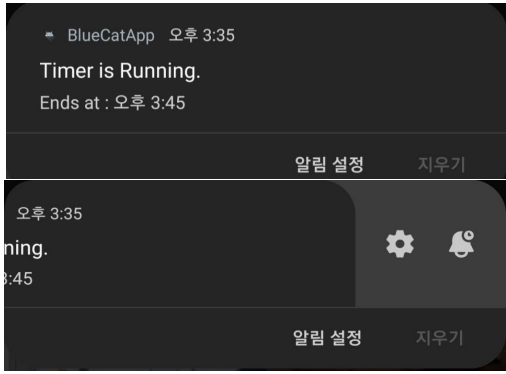


WAKE UP! WALK AROUND

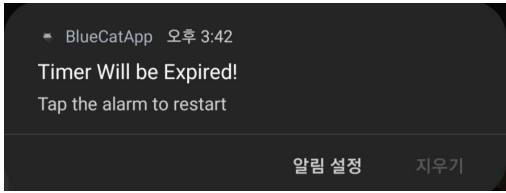
(**Vibrates** until the mission is completed & **Cannot terminate** even if the user turns off the app)



<Running>



<Pre-Notification>



Feature

Smart App-Blocking

App Blocking Use Case

Without GoldenPegasus

With GoldenPegasus

App Blocking Key Technical Challenges

Challenges	Solutions
Improving accuracy of pedometer feature	GoogleFit API step counter **LINEAR_ACCELERATION** sensor as fallback
Update fragment view on data change	notifyDataChanged to update values in adapter
Algorithms for multiple app blocking features	Different conditions to trigger each case (Differentiate HW mode and regular app blocking)

App Blocking Feature Implementations

- Notifications prior to app block
- Pedometer feature
  - Walk set amount of steps to unblock app
- Smart blocking feature
  - Detects procrastination and blocks accordingly
- Setting to disable feature

- Send toast notification 5 mins and 1 min before blocking
- GoogleFit and Accelerometer sensor to detect step count
  - Unblock when both timer and step counts completed
- Strict mode for Smart Blocking
- Toggle for features disabled when active app block exists

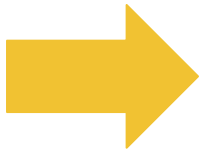
Feature

Location Reminders

From the last presentation Location Reminder

Things needed to be done

- Gamification
- Alarm or Notification
- Multiple test cases
- Various settings
- Improve UI
- Optimize power consumption



How it was conducted

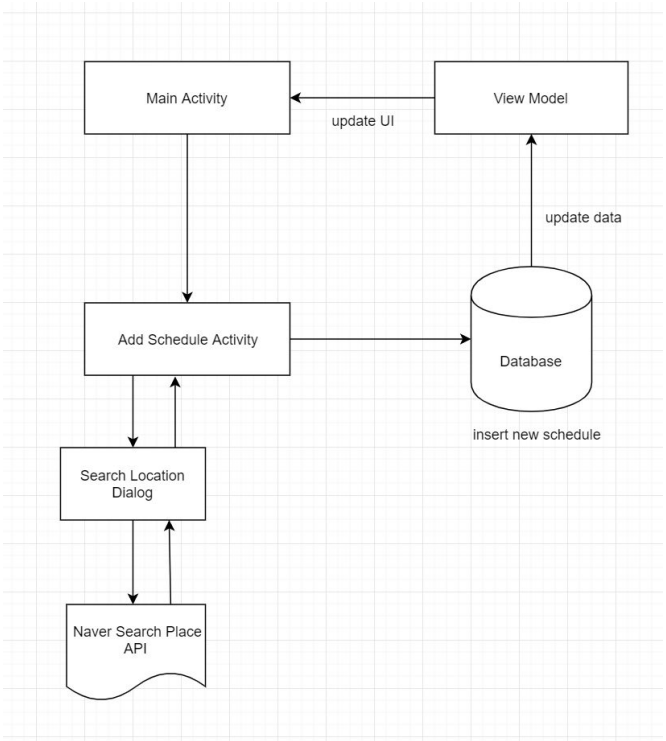
- Implemented gamification
- Implemented both
- Tested a variety of cases
- Implemented various settings
- Improved UI
- Implemented simple heuristics

From the last presentation Location Reminder

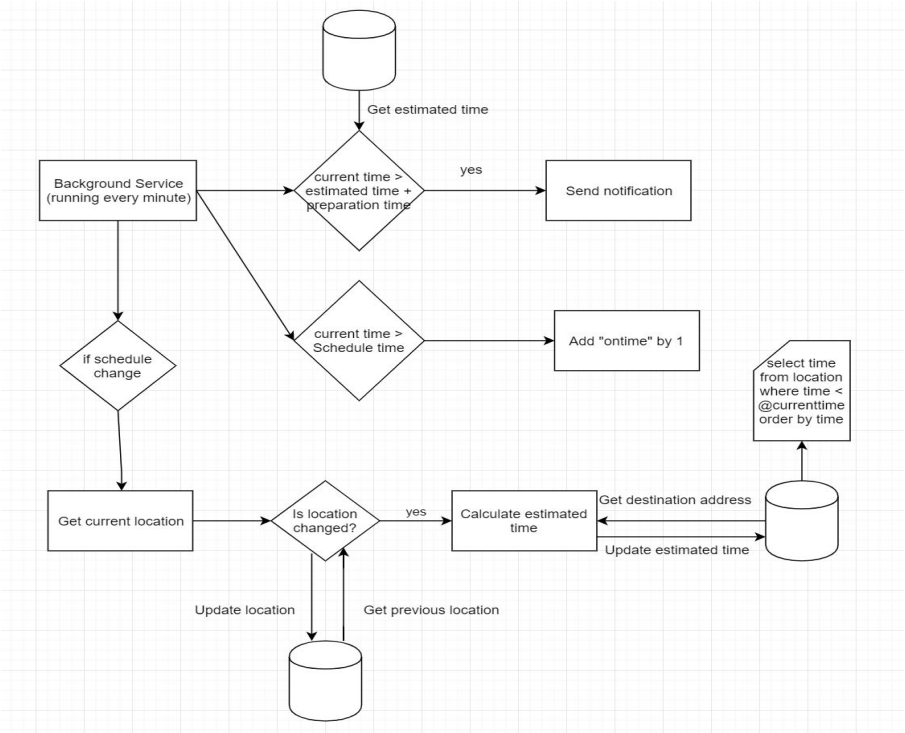
Major bug fixes

- Rapid double click issue -> lead to crash
- Bad network connection issue -> handled through try-catch
- 12 o'clock issue -> Java time function mistake
- Undesired values in database

Location Reminder System Architecture



Location Reminder System Architecture



Location Reminder Technical Challenges and Solution

Challenges	Solutions
Lots of Boundary Cases - Reboot phone after a few days - Bad network connection - Server error - Sensor error - UI issues (text overflow, etc)	**Tested a variety of cases** - Implemented checkDate() function - Implemented exception handler - Displayed overflowed text with ellipsis
High battery drain	**Minimized redundant operations** - Check location every one minute before 3 hours to the most imminent schedule **Used FusedLocationProviderClient**

Location Reminder Evaluation

- **Conducted simple experiment for battery optimization**

- Under controlled environment

  (Same duration, schedules and conditions)

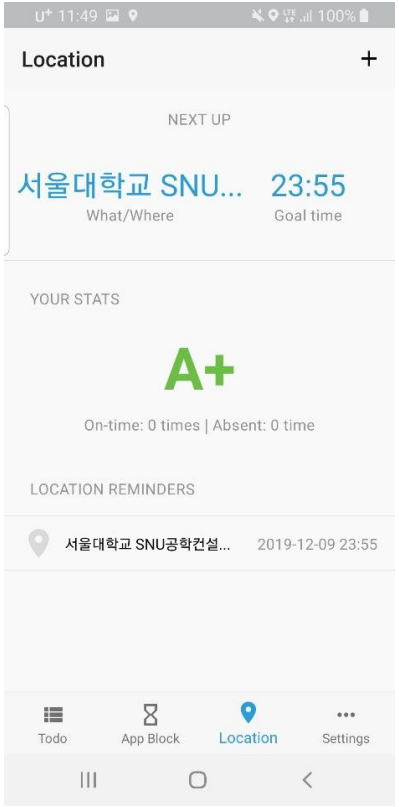
- Measured battery usage for three cases

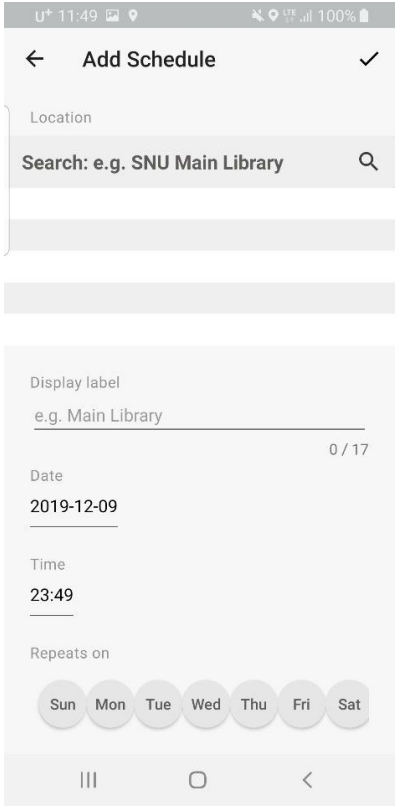
- **Experiment Results**

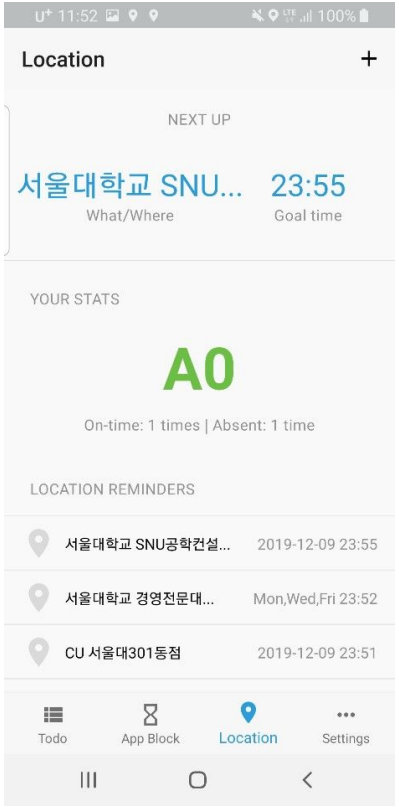
- Without our App, consumed 3% of battery
- With our App(w/o battery optimization), consumed 13%
- With our App(with battery optimization), consumed 10%

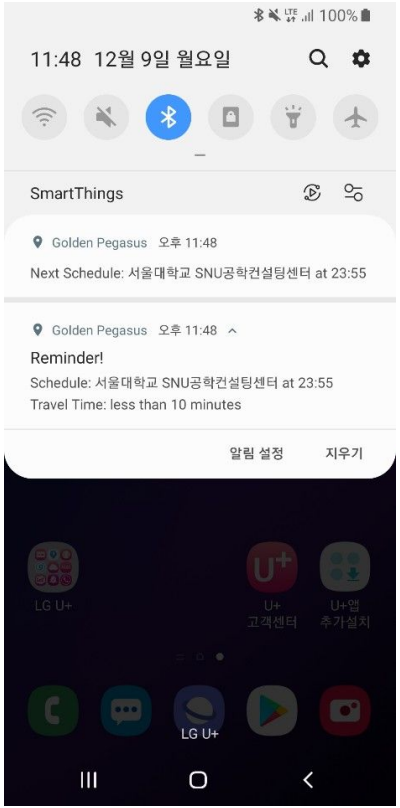
- **It works !** (Optimization would work much better for test cases with sparse schedules)

Location Reminder Demo









Project Management

Scope

Location Reminder

- Complicated Gamification → Simple Gamification
  - Simplified for easy use
- Keep estimated travel time internal → Show estimated time to user
  - Shows how long it takes to go to the destination through the notification

App Blocking

- Advanced smart blocking algorithm → Simplified algorithm based on usage & app switching
- Interactive UI (Swipe screen & selectable list items) → Simple layout, avoid over cluttering

Todo-List & HW mode

- Sending Notifications if the user has tasks due soon → No notification for such tasks
  - Too much notifications from single app

Timeline

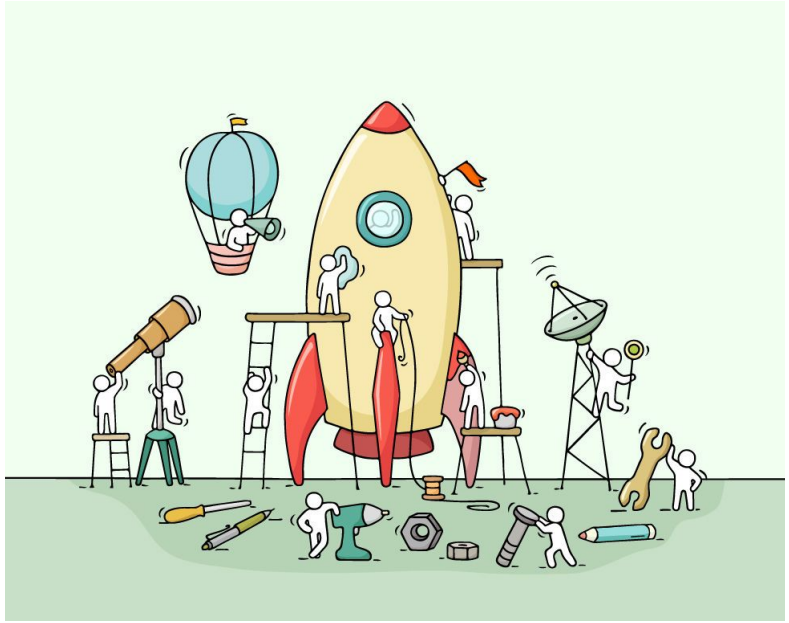
Iteration	Duration	Main Goal	Leader
1	Oct 7 ~ Oct 13	Initial planning, scheduling, mockups	Marita
2	Oct 14 ~ Oct 21	Learn Android and Kotlin, implement basic functionality	Jinyeong Kim
3	Oct 22 ~ Nov 4	Develop basic form into MVP	Seungwoo Jo
4	Nov 5 ~ Nov 18	Implement remaining	Rachel Gan
5	Nov 19 ~ Dec 2	Testing and debugging	YoungWoong Jun
6	Dec 3 ~ Dec 9	Finalize features, buffer, presentation	Marita

Roles and Contributions

1. Homework & Todo
   Jin Yeong Kim

2. App Blocking
   Rachel Gan, Sigrid Marita Kvamme

3. Location Reminders
   Young Woong Jun, Seung Woo Jo



Lessons Learnt

- (Obviously) No Plan Survives : Has been great opportunity to experience it.
- Buffers are important: Always plan ahead in case of project deviations
- Divide & Cooperate : Divided the team into small three groups based on the features and efficiently divided the tasks and cooperated when help was needed from each other.
- Conducting a lot of feature tests is important

Conclusion

Project Schedule

Efforts

- Weekly one hour meeting to catch up on project progress
- Other than the Whole Project sheet, we used Github Issues to divide/specify the tasks into small chunks

But...Deviations

- Most of the features took more time than expected to implement (tracking app usage data, tracking location , stable timer...)
- It was a great decision to have buffer week before the due date.

Lessons learned

- (Obviously) No Plan Survives : Has been great opportunity to experiencing it.
- Divide & Cooperate : Divided the team into small three groups based on the features and efficiently divided the tasks and cooperated when help was needed from each other.
- Doing a lot of test is important