

Reportal

Akshay Sunderwani, Celeste Sakhile, Maha Alidrisi, Rafif Arab, Pranay Bure, Prateek Tyagi,
Sonali Patil

Issue date: September 11, 2017

Introduction

Reportal is an application that aims to assist the Stevens community by providing an easy way for students to report issues on campus. The application will have two main interfaces, one for the students and the other is for the maintenance team.

The student's interface will allow students to report in campus issues such as, infrastructural issues, service issues, etc. through the application. They will also be able to post a picture of the damaged item, provide a description and location of the reported issue. The application will also allow students to receive alerts about the status of their reported issues. Application will be available on Android platform for Students.

As for the maintenance team, there will be a web portal which will allow the team to check reported issues and assign them to maintenance personnel. A history of the resolved and remaining issues will be maintained through this portal.

Roles and Responsibilities

- **Team Lead:** Akshay Sunderwani
- **Documentation:** Akshay Sunderwani, Celeste Sakhile, Maha Alidrisi, Rafif Arab, Pranay Bure, Sonali Patil, Prateek Tyagi
- **Documentation Editor:** Maha Alidrisi
- **Developers:** Akshay Sunderwani, Celeste Sakhile, Maha Alidrisi, Rafif Arab, Pranay Bure, Sonali Patil
- **Testers:** Akshay Sunderwani, Celeste Sakhile, Maha Alidrisi, Rafif Arab, Pranay Bure, Sonali Patil
- **Architect:** Celeste Sakhile
- **Designer:** Celeste Sakhile, Prateek Tyagi

Method

- **Software:**
 - **Languages:** Android (JAVA/KOTLIN) – Web (Angular, JavaScript)
 - **Target OS:** Android 6.0
 - **Software(Android):** Android Studio 3.0 and up
 - **Software(Web):**
 - **Code conventions:**
<https://github.com/NYTimes/objective-c-style-guide>

- **Hardware:**
 - Development hardware: Mac/Windows
 - Test hardware: Mac/Windows/Android devices
 - Target Device(Android): Device with Android 6.0 and up
 - Target Device(Web): Web browsers (Chrome, Firefox)
- **Review Process:**

Code reading: reading each other's code every week.

Continuous integration: the team will meet every week to integrate their parts and resolve issues.
- **Build Plan:**

GitHub will be used for code collaboration and Trello for sharing project related work and managing user stories.

 - **GitHub link:** https://github.com/akshya672222/SSW695_Team7
 - **Trello link:** <https://trello.com/b/M7BJNTxQ/ssw695>

Virtual and Real Work Space

Trello will be used as a work space to share project related and details.

- **Trello link:** <https://trello.com/b/M7BJNTxQ/ssw695>

Communication Plan

- **Scrum/Status Meetings**
 1. Scrum meeting will happen on every Thursday 11AM - 2PM at Altorfer or any other place depending on availability.
 2. Make sure to updated the meeting notes from the scrum meeting outside of class and meetings.
 3. When work is finished, and milestones completed, strikethrough the requirement line to show progress, ~~like this~~.
 4. You should write a completed by: for each strikethrough.
 5. Your requirements and intended features may change during scrums. Make sure you update this Readme if that happens (I support active change in repo).
 6. Make sure you attend meet regularly and show me the completed requirements in meet.
 7. Team members can work in meet and at home as well but they need to make sure that their part of work is done before meet.
- **Additional Meetings**

If required, the team will plan to meet on campus and work together to resolve issues and meet the deadlines.

Timeline and Milestones

Week #	Tasks
Week 01	Create initial development plan, draft architecture
Week 02	Finalize the architecture, paper prototype, database design
Week 03	Design UI for app, integration, create database

Week 04	Design UI for app, integration
Week 05	Set up webserver, UI design for web portal
Week 06	Backend development, API development, integration, testing
Week 07	Backend development, API development, integration, testing
Week 08	Backend development, API development, integration, testing
Week 09	API integration, testing
Week 10	Final testing
Week 11	Final demo

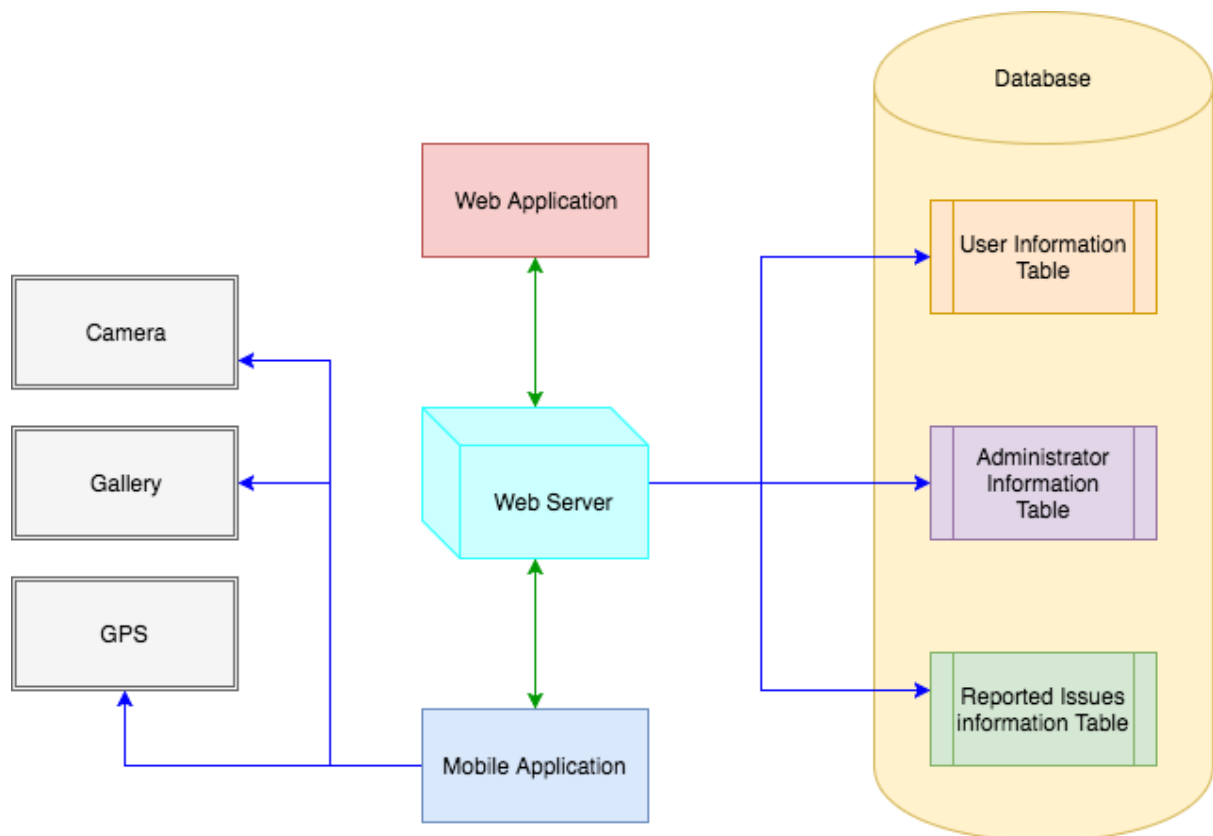
Risks

- Secure data storage to avoid exposing key attributes like user-id, passwords, authentication tokens, location data etc. as plain text.

Assumptions

- Reportable application will need access to Location, Photos/Media, Camera, WIFI features of user's android smartphones for app to operate
- Application may not function slow in case of server issues
- Application will behave similar on Android versions 6+

Software Architecture



Distribution list

1. Team
2. Manager/Professor Gregg Vesonder