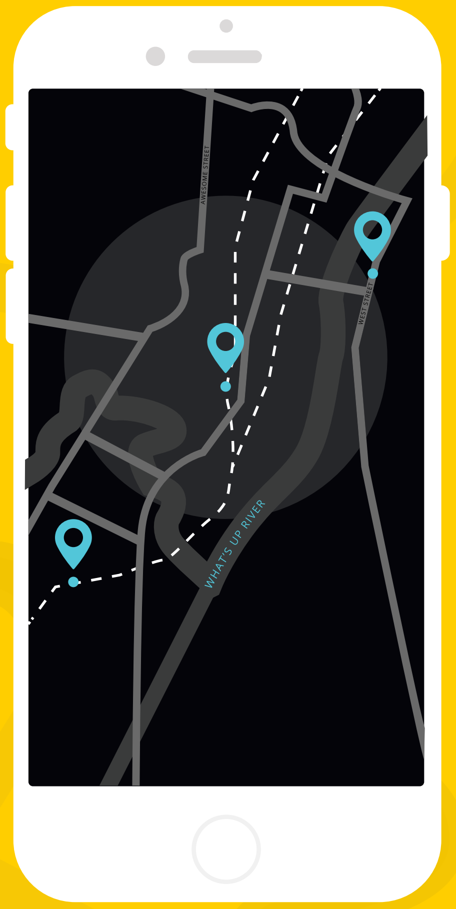


EMPLOYEE LOCATION TRACKING SERVICE

Web & Android

mobiversal

www.mobiversal.com

OVERVIEW

GPS fleet tracking services have been on the market for some years now but with the explosion of smartphone usage, apps became a better option to track employee location.

We've compiled this proposal for a service that would help companies increase the safety, efficiency, and productivity throughout their operations. This is a full-featured platform where companies can signup and start benefit from employee tracking through mobile apps. The tracking system consists of a web-based dashboard controlled by a manager and mobile apps installed on employees smartphones that communicate with the central console.

Some of the advantages of using the employee location tracking service:

- the administrator can see where all workers are, if they are on schedule and
- can dispatch the closest worker to a certain job
- businesses can give clients accurate and complete information about their
- product delivery or when a technician will arrive at their location
- employees can receive turn-by-turn driving directions to find the best way to
- the client or to locate the nearest restaurant or gas station
- the apps can serve as a communication channel saving companies a lot on phone bills
- employees have to fill less papers as they can use the app to clock in or clock out – businesses can set up certain alerts when employees are entering or leaving an area using geofencing

Businesses that can benefit from personnel tracking apps include construction, delivery, field services, fleet leasing, utilities, food and beverage, oil, mining companies and many more.

ABOUT MOBIVERSAL

Mobiversal is a mobile app development company working with startups and established brands, like Forbes, to create unique mobile experiences. Since its beginning in 2011, Mobiversal has been focused solely on developing iOS and Android apps. The company was ranked among Europe's "Top Mobile App Developers" (Clutch.co) and has turned ideas into apps for over 60 clients from more than 15 countries.

OUR PROCESS

Discovery

We work with clients to understand their business and we merge their initial concept with everything the Mobiversal team knows about mobile. We'll assess the possible challenges and identify the ways to overcome them.

Features & Architecture

We establish what features go into the product and how they will work together. Here, we're drafting a skeletal framework for the app in the form of wireframes.

Design

When it comes to first impressions, it's all about design. We'll put our passion for good design to work and based on approved wireframes we'll design all screens.

Development

The development process is broken down into sprints based on feature sets. Our agile-based development process will allow you to regularly review and assess what we're building.

Quality Assurance

Our Quality Assurance team will test the app after each development sprint and once all major functionality is implemented, we'll prepare a Beta Build.

Launch

After passing the Beta Build through a final round of QA and refinements, we'll have in our hands a Release Candidate Build. We can either submit the app to the App Stores or provide you with everything you need to do it yourself.

Maintenance

We provide our clients with a 3 months bug fixing period, free of charge, and maintenance packages that cover everything from small updates all the way to whole new versions.

APPROACH TO MEET REQUIREMENTS

The service will have several components, each for a different type of user:
• the web dashboard for companies (fleet controllers)
• the mobile app for employees
• the admin area for the service owner

In order to build this system we will need:
- design work for the website (Service presentation website + Web dashboard)
- for Companies (fleet controllers)
- website coding
- design work for the mobile app
- mobile app coding
- admin area coding
- backend (server side) coding

Design

1. Identity. Logo, app icon, colour scheme, splash screens
2. Website design. Service features presentation + Company (fleet controller) web dashboard
3. Mobile app design

Backend

The backend is the database and code “invisible” to the user -- that is, code that runs on the server for storing and retrieving the data used by the apps. The backend consists of an API, a set of network calls used for connecting the web and mobile clients with the database.

1. Architecture and structure of entire project.
2. API documentation and implementation
3. Setup server hosting.

Website

The website (web client) is used by companies (fleet controllers) to manage employee information, assign tasks to employees, track their location and view reports.

1. Home screen. Will present the tracking system, features, prices.
2. Register as a company (fleet controller)
3. Login
4. Company (fleet controller) profile
  4.1. Company information
  4.2. Subscriptions and payments
5. Map with employees
6. List of employees
7. Add new employee. Fill a list of employee settings and generate a code that will be shared with the employee to sign up.
8. Edit employee settings
9. List employee tasks
10. Add employee task
11. Employee reports
  11.1. Daily
  11.2. Monthly
  11.3. Custom period
12. Map with zones + create zone where the employees are auto notified when entering.

Mobile app

The mobile app is used by the employees for getting new tasks, getting directions to the destination as well as reporting the tracks.

1. Login/Register device. Enter the code received from the Company (fleet controller) to start the app
2. My profile. The employees can see their profile, without the possibility to edit (only the controller can update employee details).
3. List of tasks
4. Task details:
  4.1. Start/end task
  4.2. Task estimated plan
  4.3. Directions to destination
5. Receive alerts for certain zones

Website

The admin area is used by system administrators to manage information about companies and see some general reports. The companies (fleet controllers) are the users of the website described above.
1. Login
2. List of companies
3. Company details
  3.1. Company information
  3.2. Employee and tasks statistics
  3.3. Fleet invoices
4. All invoices

TECHNICAL SOLUTION

We are using RUP (Rational Unified Process), so that the application modules can be developed in parallel. The diagram in the “timeline and costs” section describes the overlapping of the development for these modules.
Technologies proposed for the development:

1. Backend and Database

We have selected a Javascript-based stack to implement the backend, based on NodeJS, ExpressJS and SailsJS. As opposed to more traditional technologies, Javascript is a newer one but growing quickly on the server-side, and we selected it because of its stellar performance and better scalability.

The database storing employee location data will be implemented as a NoSQL database, as that data grows quickly and a traditional SQL database will not provide support for the needed scalability. The database system will be MongoDB as it has extended support for location processing (spatial indexing), and additionally it integrates smoothly with NodeJS.

2. Admin Area and Website (web client)

We are using AngularJS 2.0, a modern Javascript-based framework to implement the website. The main screen will display real-time locations of all the employees using Google Maps.

The admin area will have a simpler UI and functionality and will be implemented using a template based on Bootstrap 3.0.Both the website and the admin area will run on all major current browsers: Chrome, Firefox, Internet Explorer.

3. Mobile app

The mobile client application will run on all phones with Android 4.0 or newer that include Google services. It will be developed using the native development tools provided by Google and will use Google Maps for some features (directions to destination). The app will run in portrait mode and will have a common layout and look-and-feel across all form factors (phones and tablets).

The app will remain running when in background and will report the user's location periodically to the server using a simple schedule.

A smarter algorithm may also be developed, to report fewer locations but still keep a good accuracy in order to minimize battery and network usage. This would be an extra feature, outside the scope of this proposal.

During the development phase we propose at least the following:

- Each week we are delivering demo files (images, web-pages or mobile demo builds)
- Each week we must have a general sync meeting to discuss:
  - last week's progress based on the demo provided
  - next week plans
  - other items regarding that period

Tools and technologies used:

- IDEs (code editors): Android Studio for Android, WebStorm for web, backend and admin area
- Programming Languages: Java for Android, Javascript and HTML for web, backend and admin area
- Frameworks and 3rd party SDKs: ExpressJS, SailsJS for the backend,
- AngularJS, Bootstrap for the web client and admin area, Google Maps for the mobile app as well as web client area.
- JIRA for issue ticketing and project management
- Email and Google Drive for sharing documentation, specifications and content
- Fabric by Twitter for crash reports (Crashlytics), beta distributions (Beta) and analytics (Answers)

TIMELINE AND COSTS

Before starting the actual designing and coding, we will need to build a full functional requirements document. This document will contain everything the system will be able to do with higher level of details, because it will be the way to check and compare everything needed to be done. The designer, software architect, software developer, QA team, product owner, and project manager will know exactly what to build and what to expect from the system. This part should not take more than 1 week and will require communication between our technical business analyst and the product owner.

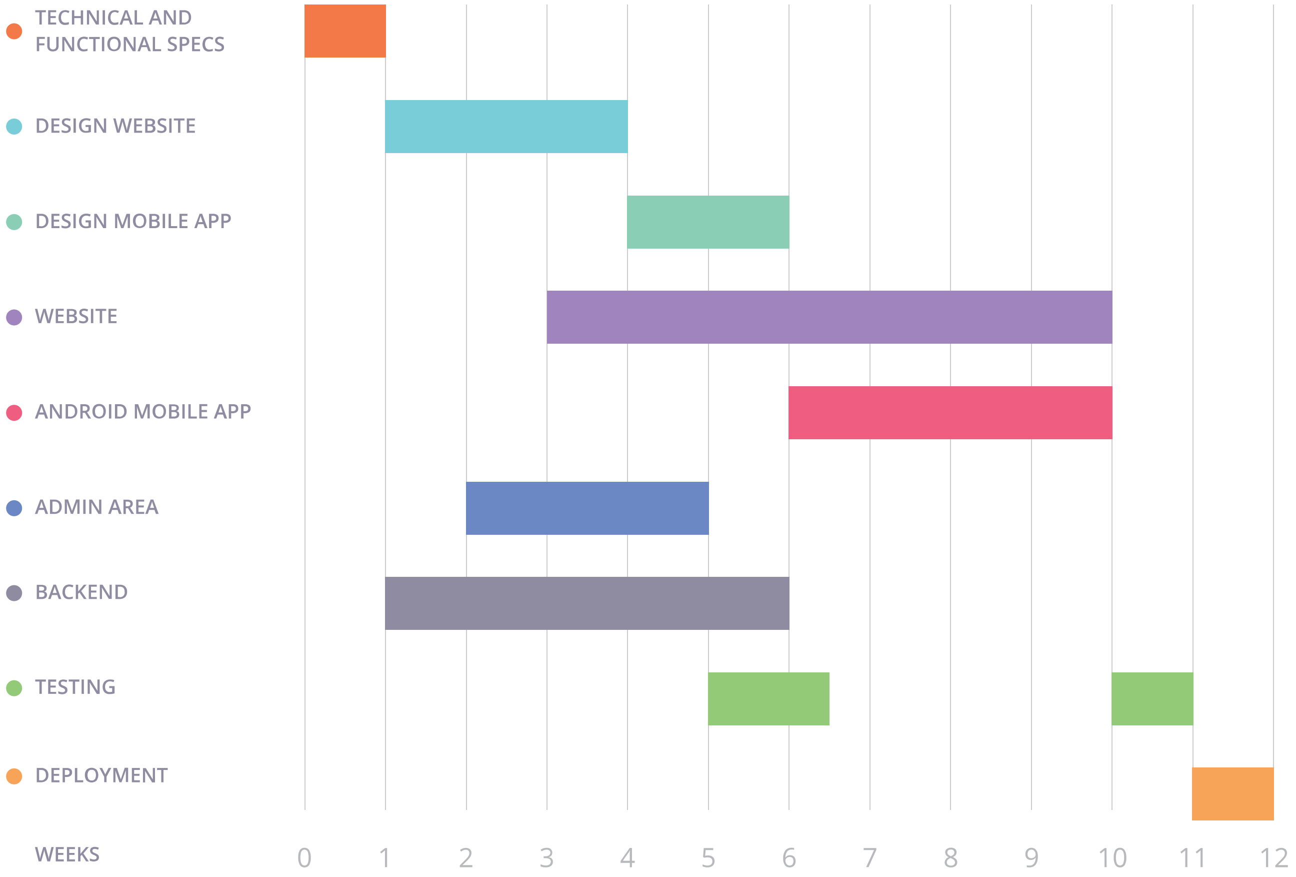
After this scoping week, the designer and the software architect can start working. After having the architecture, the developers will also be able to start developing.

During the development we will have two important milestones: alpha milestone and beta milestone. For the alpha milestone, around 40-60% of the features should be working and for the beta milestone 100% of the features should be working, but with chances of bugs.

After the beta milestone, the apps will be sent for QA testing and we'll fix any found bug. After fixing this bugs, the apps will be ready for your user acceptance testing. After the user acceptance testing is ready (which means you've tested and found that everything is ok), it starts a warranty period of 3 months.

The total estimated time of development is between 11 and 13 weeks. Below you can find a timeline chart for the whole project, with each component apart.

TIMELINE



COSTS

COMPONENTS	DEVELOPMENT PERIOD
Design mobile app	2 weeks
Design website	3 week
Android mobile app	4 weeks
Website	7 weeks
Admin area	3 weeks
Backend	5 weeks
Project management	70 hours
QA Testing	2 weeks
TOTAL	**11 - 13 weeks**
	TOTAL: $28.000

Do you want to have this project done by Mobiversal or do you have a new challenge for us?

We love taking ideas and turning them into real apps. Tell us your idea and we'll give you details about costs. Simply click below to tell us more about your project.

Get a free quote →