Group 27

Green Core

The Smart Gardner

G27

Team

Supervisor

Dr Dinuni K Fernando

Co-Supervisor

Ms. Sithara Fernando

Group Members:

M.M.H Tharushika	17001765
H.U.K.M Pabasara	17020611
J.A.N.C Niroshana	17001196
B.A Medawatta	17001072
K.C Gamage	17000475
K.S.A Ahamed	17020034

Outline

- Introduction
- Problem Definition
- Goal and Objectives
- Project Scope
- Deliverables
- Functional Requirements
- Quality Attributes
- Feasibility Study
- Proposed Technologies
- Project Delivery Timeline
- High Level System Architecture
- System Development Methodology

Green Core - The Smart Gardner

Outline

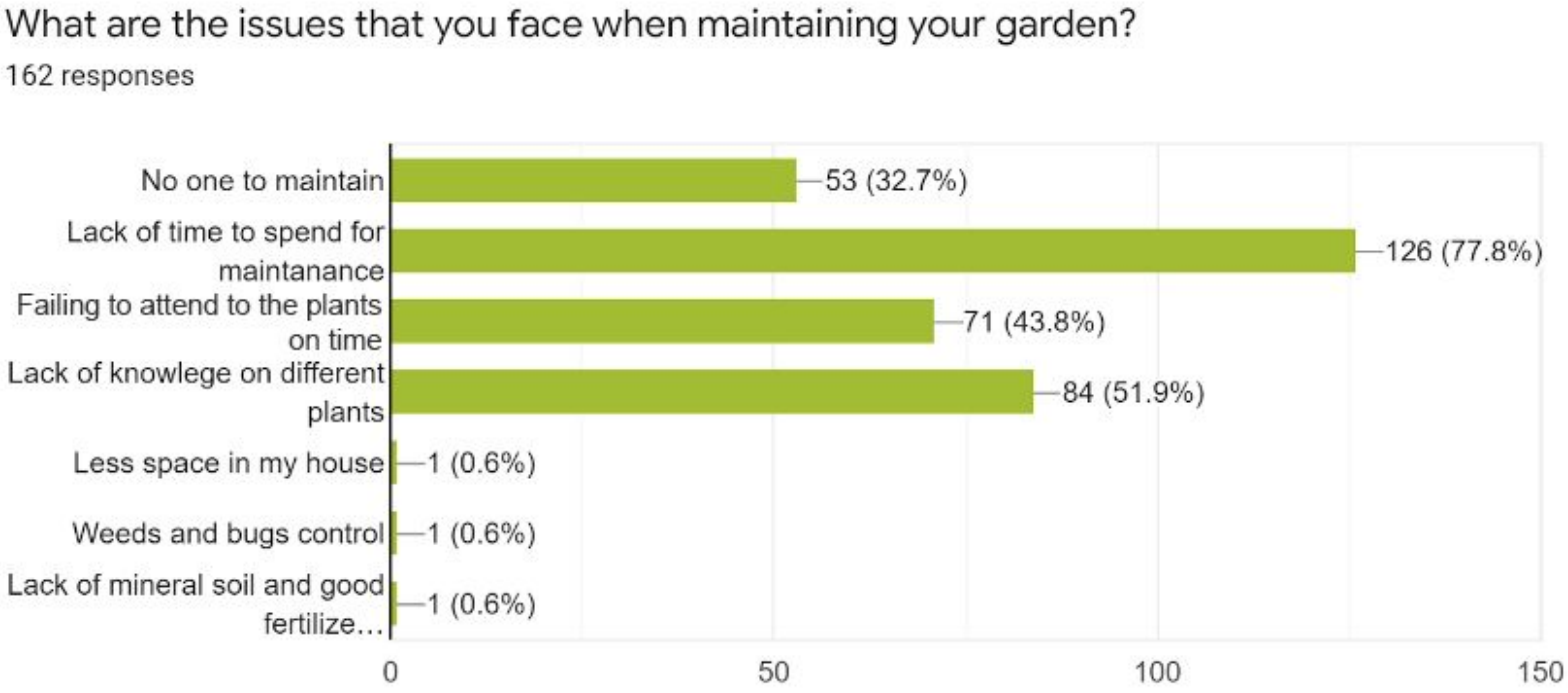
- Introduction
- Problem Definition
- Goal and Objectives
- Project Scope
- Deliverables
- Functional Requirements
- Quality Attributes
- Feasibility Study
- Proposed Technologies
- Project Delivery Timeline
- High Level System Architecture
- System Development Methodology

Green Core - The Smart Gardner

Problem Definition

- People tend to stay at home leisurely for limited time Due to their busy lifestyles.
- Gardening with minimum effort is worth and effective.
- With the pandemic which occurred in recent past, people tend to grow their own produce for sustainability.
- But after this situation people may not have time to take care of their plants due to their busy lifestyle.

Results of the survey



Problem Definition

- People tend to stay at home leisurely for limited time Due to their busy lifestyles.
- Gardening with minimum effort is worth and effective.
- With the pandemic which occurred in recent past, people tend to grow their own produce for sustainability.
- But after this situation people may not have time to take care of their plants due to their busy lifestyle.

Introduction

- Green-Core is an automated gardening system consists of both manual and automatic gardening features.
- Enable users to control their garden while they are away from home by using a smartphone.
- Capable of real gardening features such as watering plants.
- System consists of an IoT device and a mobile app for the user and a web app to control inner settings.

Outline

- Introduction
- Problem Definition
- Goal and Objectives
- Project Scope
- Deliverables
- Functional Requirements
- Quality Attributes
- Feasibility Study
- Proposed Technologies
- Project Delivery Timeline
- High Level System Architecture
- System Development Methodology

Green Core - The Smart Gardner

Goal and objectives

Our Goal

Our main goal is to develop a smart garden environment that allows the user to provide a real-time gardening experience via a mobile phone.

Objectives

- People can get fresh vegetables and food from their garden with aid of a mobile application.
- Track the progress of each plant's status.
- Adjust the growing environment remotely and automatically manage the garden lively.
- Get real-time notifications of individual plant status via mobile App.
- Administrator can collect, view data of all gardens via web application and use them for identifying vulnerable units and predictions, etc.

Project Scope

Project Scope

In Scope

- Interactive Mobile app and an IoT device for the users
- Web App for the system admin
- System will be hosted in cloud
- System will be able to operate centrally

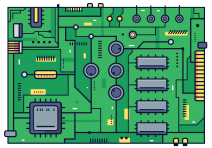
Out Scope

- Train a machine learning model using the data gathered by the sensors to predict optimal conditions to grow specific plant types.
- Generating reports regarding useful analytical data.

Deliverables

- Mobile app

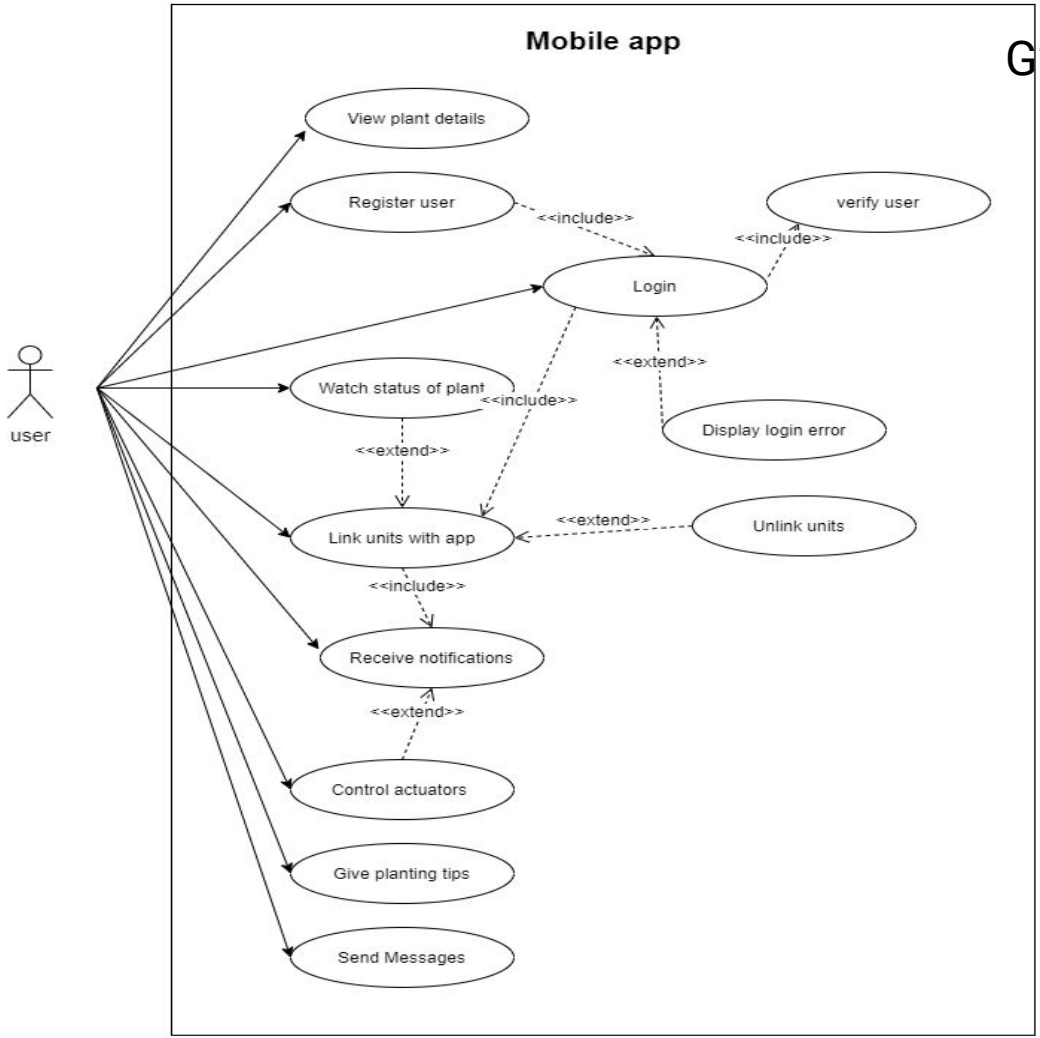
- IoT Device



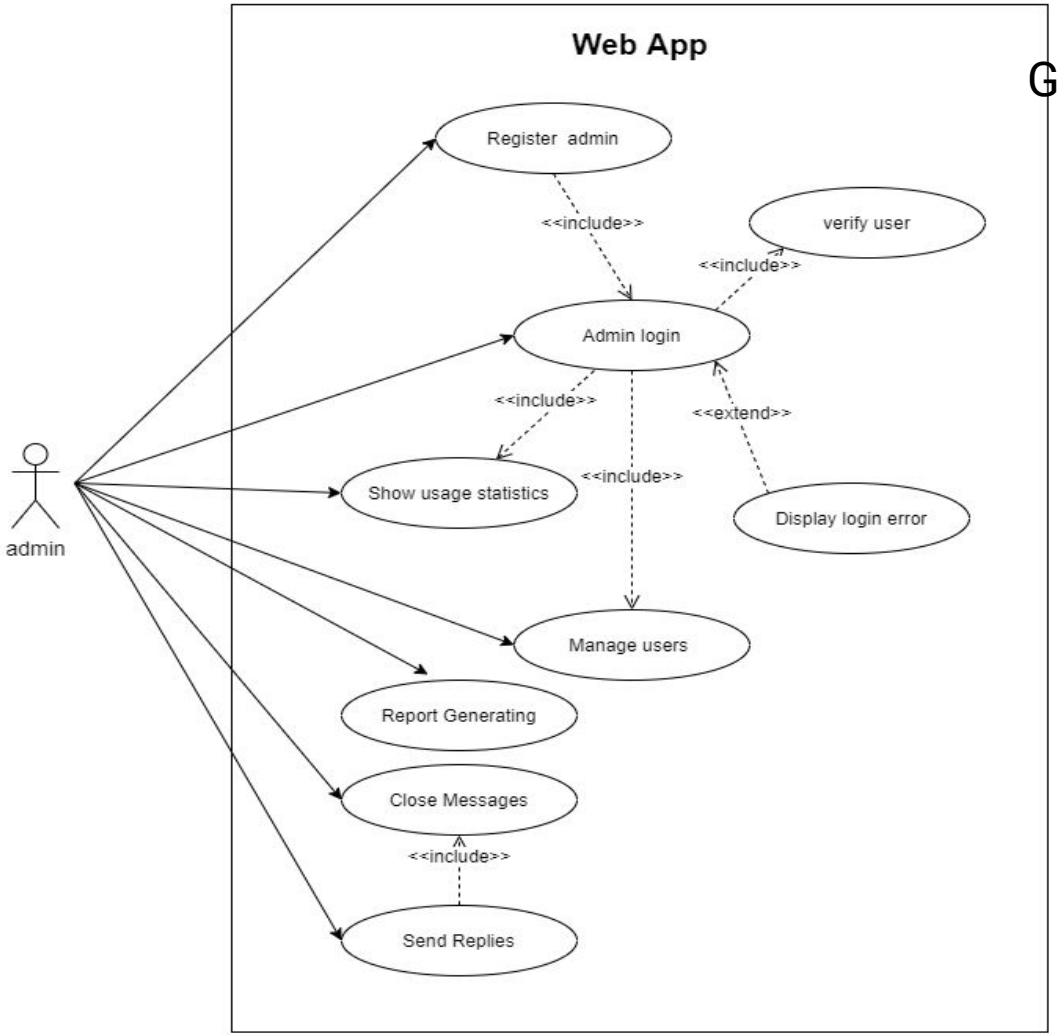
- Web App

Use Case Diagrams

Mobile Application



Web Application



Outline

- Introduction
- Problem Definition
- Goal and Objectives
- Project Scope
- Deliverables
- Functional Requirements
- Quality Attributes
- Feasibility Study
- Proposed Technologies
- Project Delivery Timeline
- High Level System Architecture
- System Development Methodology

Green Core - The Smart Gardner

Functional Requirements

Mobile Application

- User can sign up and log into the mobile application and setup account settings.
- Application is able to get feedback of IoT units.
- Real time notifications regarding status of the user's garden.
- Able to give planting tips to the user.
- User can monitor the garden status.
- User can add new units/modules
- Users can contact administrators using the inbuilt chat .

Web Application

- User(admin) can sign up and log in.
- User can update profile details.
- View analyzed data(graphs) about gardens.
- View local map of all the mobile app users garden location.
- Notify about failed or vulnerable units.
- Contact users regarding their issues and issues with their IOT devices(Chat Module).

Quality attributes

Availability

System will be available on both mobile and web with any internet accessible device.

Usability

Simple and easy to use interface throughout the mobile and web application.

Dependability

Backups backend database to a database in the cloud periodically.

Security

Only administrators can access the web application and only registered users can enter to the mobile app.

Assumption- Administrator is a trustworthy person.

Maintainability

Mobile app can be updated easily through Playstore.

Repairs on the IoT device can be done easily because of the modular nature of the IOT device.

Outline

- Introduction
- Problem Definition
- Goal and Objectives
- Project Scope
- Deliverables
- Functional Requirements
- Quality Attributes
- Feasibility Study
- Proposed Technologies
- Project Delivery Timeline
- High Level System Architecture
- System Development Methodology

Green Core - The Smart Gardner

Feasibility Study

Technical Feasibility

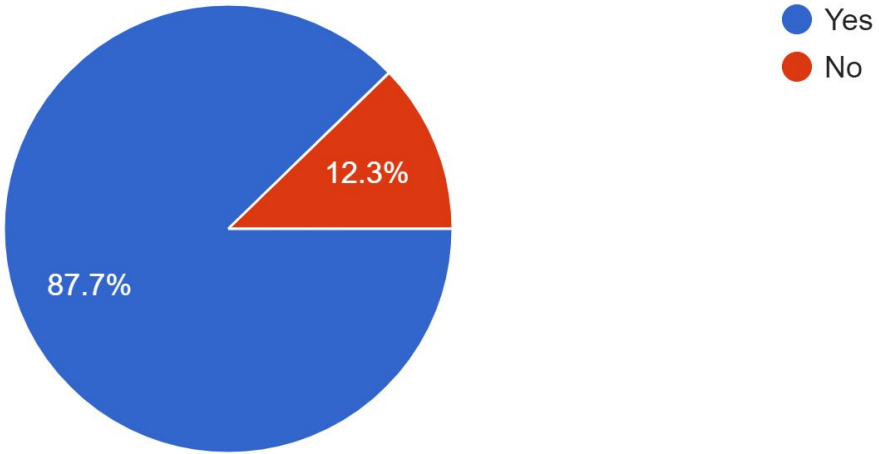
- To satisfy our requirements, IoT device ESP8266 and arduino mega is sufficient along with some sensors such as soil moisture, humidity, etc.
- We use Amazon Web Server(AWS) which is also available for free for a year to host our virtual servers.
- MongoDB , Node.js and Reactjs are Free and Open Source.
- Mobile application will run on any tablet or smartphone with proper backups.
- Web application will run on all major browsers with proper security features.

Operational Feasibility

- Both the mobile application and web application will provide simple and intuitive Interfaces to the user.
- Web based System so that user can enroll with the system at any time any where.
- Almost everyone uses smartphones in their daytoday life so an app is a good way to get notifications about the plantation.

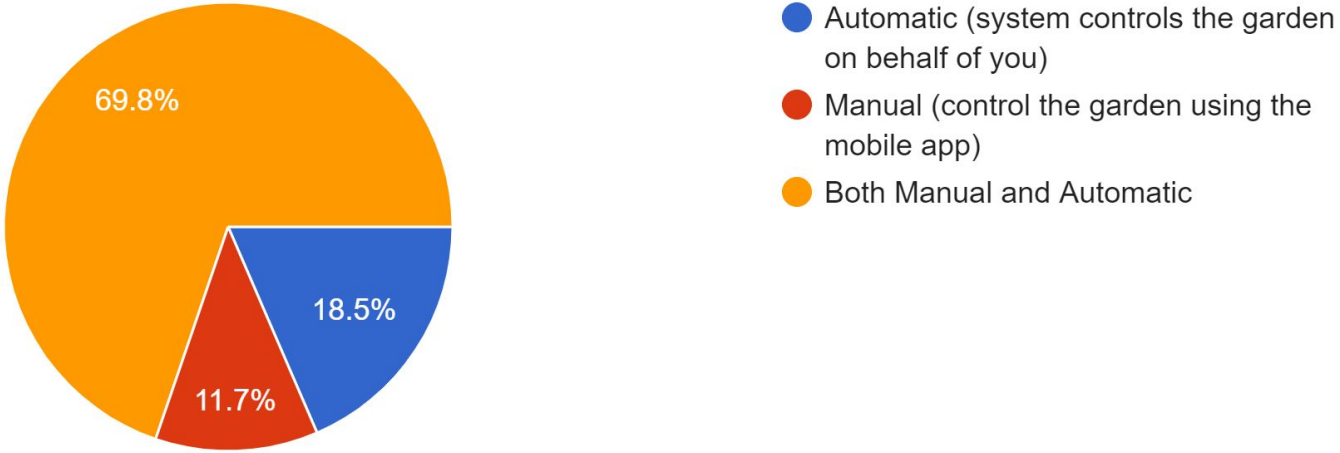
Do you prefer to use an IoT device and a mobile app to control your garden?

163 responses



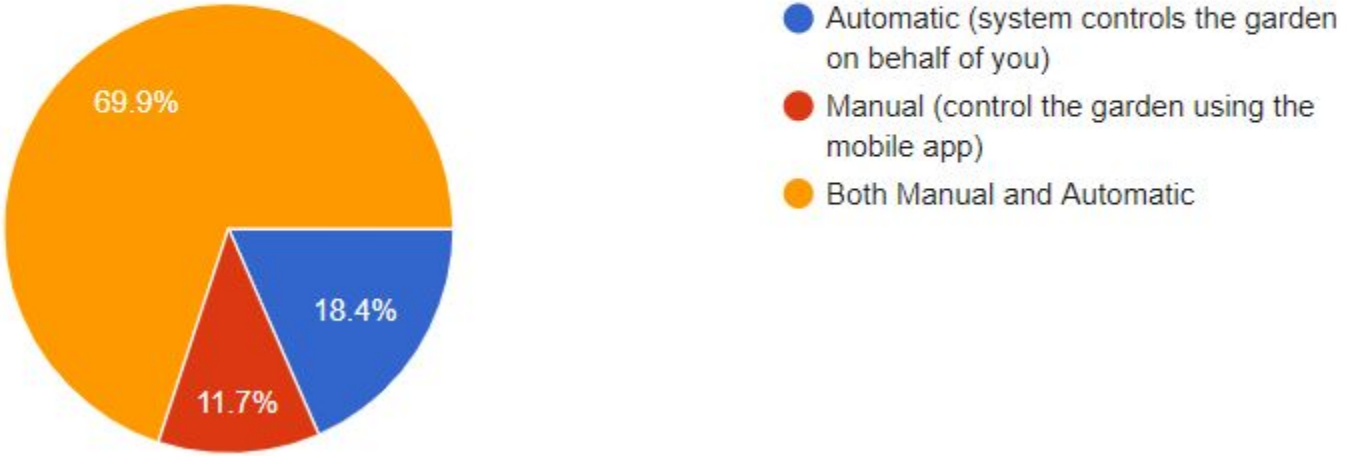
Which type of system do you prefer?

162 responses



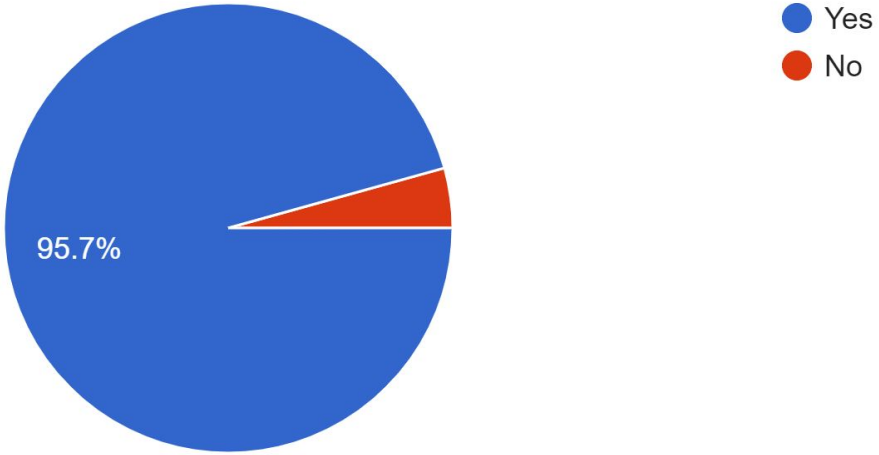
Which type of system do you prefer?

163 responses



Is it comfortable for you to get the service of this type of application with the facilities you already have?(WiFi, Mobile devices etc)

163 responses



Economic Feasibility

- The web application is hosted in the AWS free-tier server, which is provided by AWS for free for one year and this bears zero cost for the first year of operation.
- As we use MongoDB for our database which is free and there are 25 Million database operations provided for the free tier.
- We develop IoT device using arduino uno and ESP8266 as they are cost effective (cost will be around 20$ per unit)
- Since it is developed only by a team of university students, development cost also can be factored as none.

Schedule Feasibility (Time Estimation)

- Number of work hours for a member
  - Weekdays = 3*5
  - Weekends = 4*2
- Man-hours per week = (3*5 + 4*2)*6= 138
- Estimated number of weeks = 13
- Estimated total of man hours = 138*13 = 1794

Legal and Ethical Feasibility

- Privacy of information - User details and their garden information will not be exposed at any cost.
- Handling user data - The interest of the user within the system is confidential and will not be sold or used for advertising purposes (AdSense etc.)
- User and IoT device location will be recorded only with the consent of the user.

Outline

- Introduction
- Problem Definition
- Goal and Objectives
- Project Scope
- Deliverables
- Functional Requirements
- Quality Attributes
- Feasibility Study
- Proposed Technologies
- Project Delivery Timeline
- High Level System Architecture
- System Development Methodology

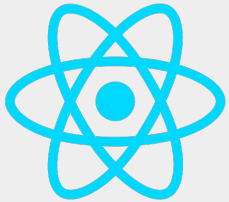
Green Core - The Smart Gardner

Proposed Technologies

Mobile Application

React Native

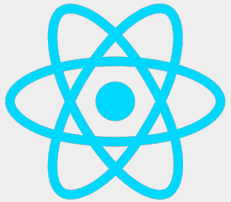
- Easy to learn.
- Cross platform mobile apps from a single codebase.
- Higher performance .
- Component based development.
- Native look and feel.



Web Application

React JS

- UI could segmented into components and write logic accordingly.
- Has a shallow learning curve.
- Could reuse components.
- Syntax is much like HTML with javascript.



Web servers

NodeJS with Express JS

- Easy to scale the applications horizontally as well as vertically.
- Higher performance because of non-blocking I/O operations.
- Easy to learn because of the use of javascript .

Database

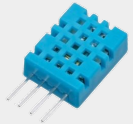
MongoDB (NoSQL)

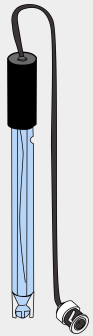
- Due to the structuring (BSON format - key value pair) way of the data in MongoDB, no complex joins are needed.
- Processing and analyzing big data is easy with NoSQL.
- Scalability and Availability requirements for sensor data storage.
- Easy to scale up when the system grows (long term).
- Clustering - can have different clusters for different regions (long term).

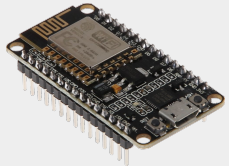
IoT device

Arduino Mega and ESP8266

- Production cost is low. we can print our own Arduino board by customising according to the requirement.
- Arduino is open source.
- Documentation is available for free.







ARDUINO™

Outline

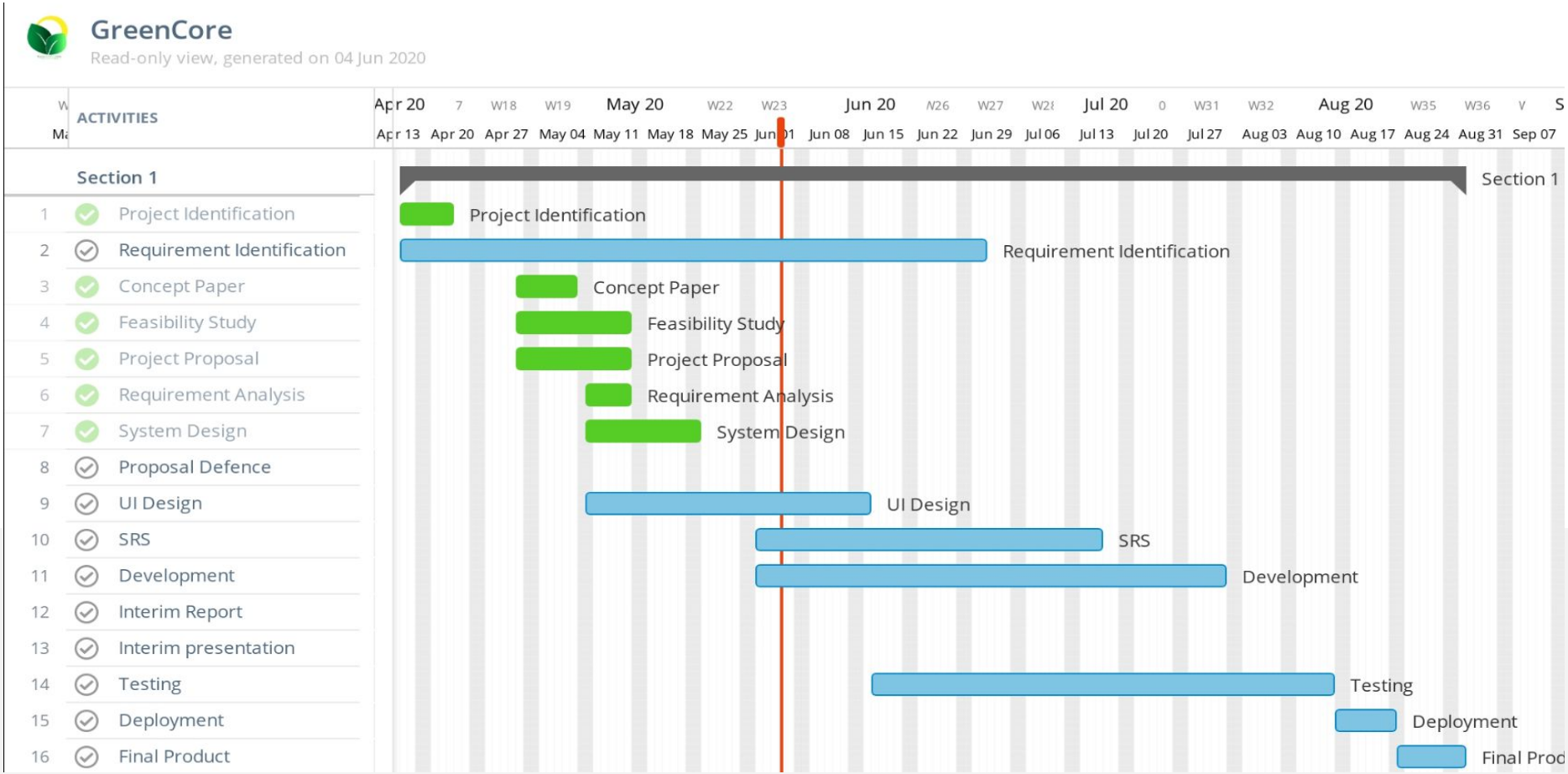
- Introduction
- Problem Definition
- Goal and Objectives
- Project Scope
- Deliverables
- Functional Requirements
- Quality Attributes
- Feasibility Study
- Proposed Technologies
- Project Delivery Timeline
- High Level System Architecture
- System Development Methodology

Green Core - The Smart Gardner

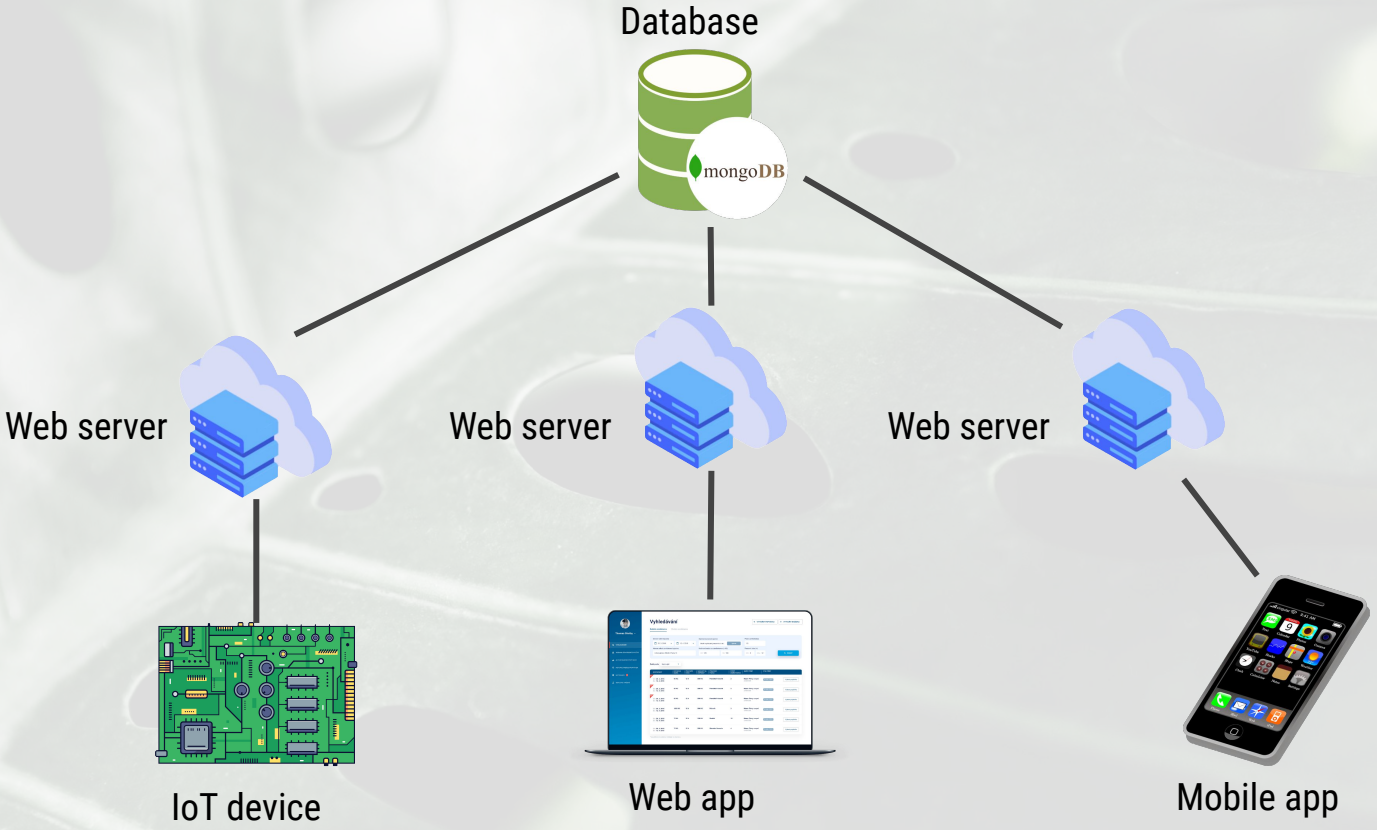
Project Delivery Timeline

Project Delivery Timeline

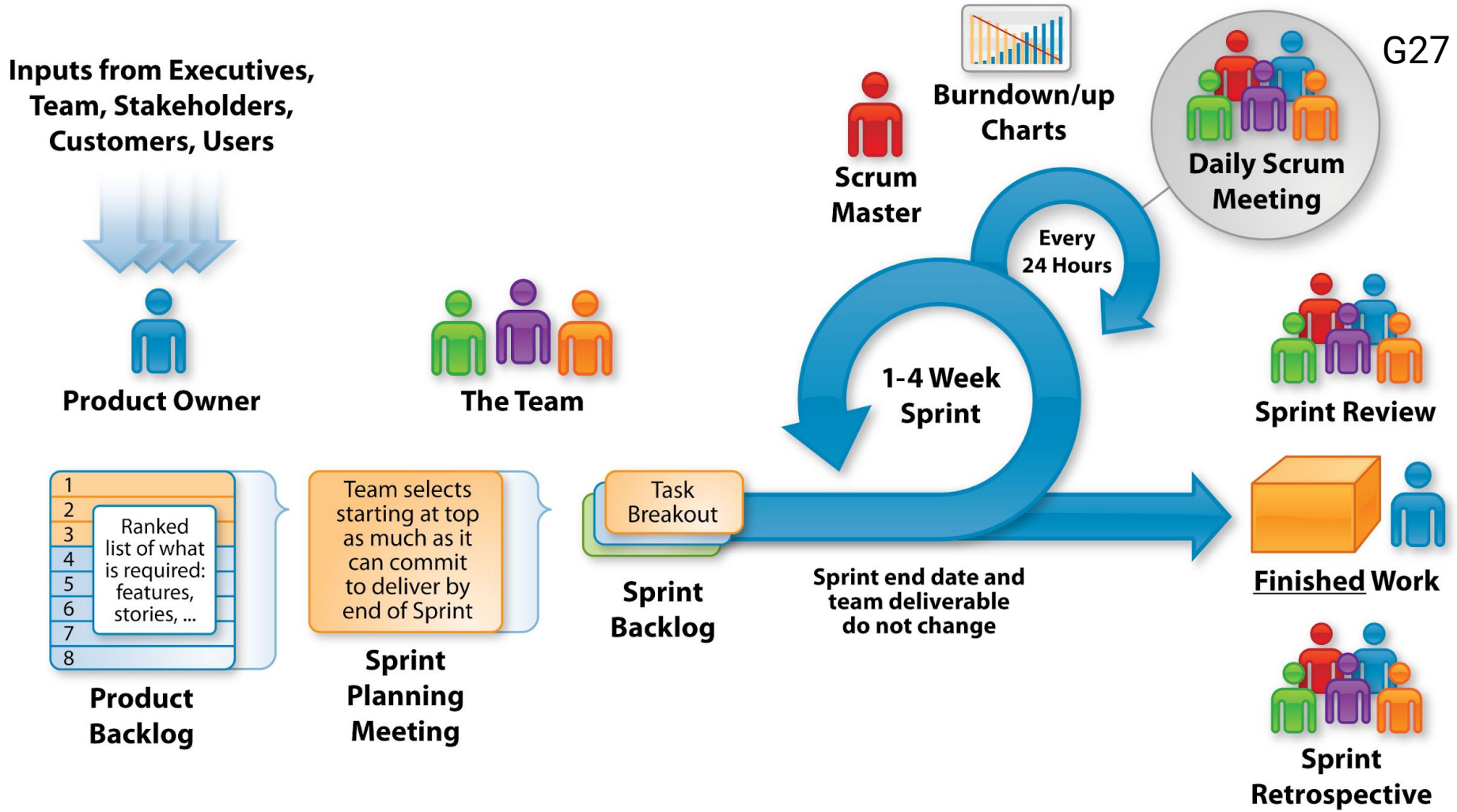
G27

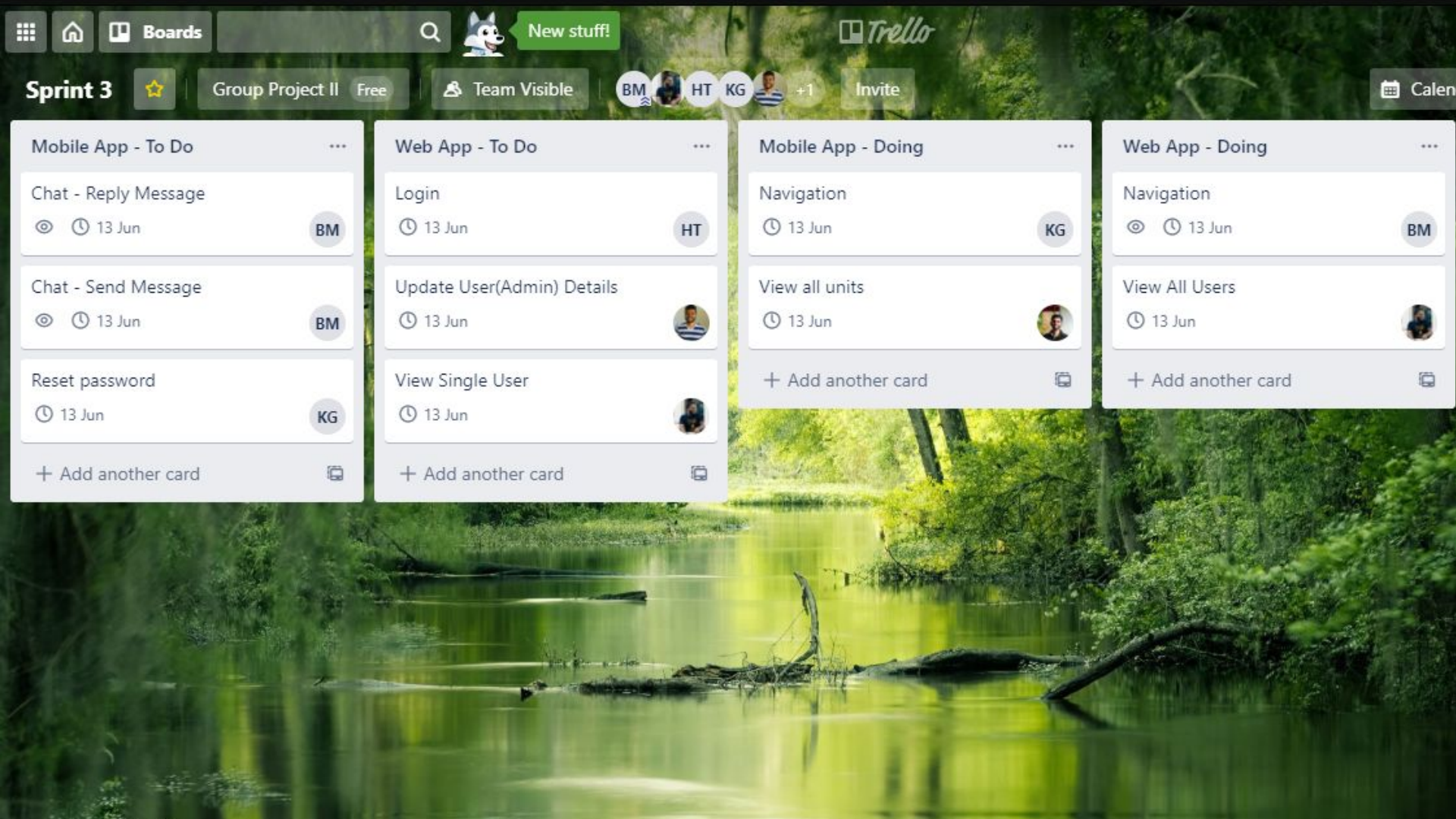


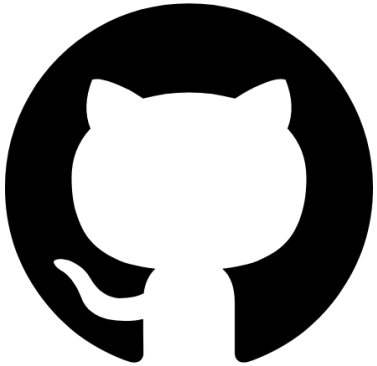
High Level System Architecture


Database
mongoDB
Web server
Web server
Web server
IoT device
Web app
Mobile app

System Development Methodology & Project Management


Inputs from Executives, Team, Stakeholders, Customers, Users
Product Owner
The Team
Scrum Master
Burndown/up Charts
Daily Scrum Meeting
Every 24 Hours
1-4 Week Sprint
Sprint Review
1
2
3
4
5
6
7
8
Ranked list of what is required: features, stories, ...
Product Backlog
Team selects starting at top as much as it can commit to deliver by end of Sprint
Sprint Planning Meeting
Task Breakout
Sprint Backlog
Sprint end date and team deliverable do not change
Finished Work
Sprint Retrospective


Boards
New stuff!
Trello
Sprint 3
Group Project II
Free
Team Visible
BM
HT
KG
+1
Invite
Calen
Mobile App - To Do
Chat - Reply Message
13 Jun
BM
Chat - Send Message
13 Jun
BM
Reset password
13 Jun
KG
Add another card
Web App - To Do
Login
13 Jun
HT
Update User(Admin) Details
13 Jun
View Single User
13 Jun
Add another card
Mobile App - Doing
Navigation
13 Jun
KG
View all units
13 Jun
Add another card
Web App - Doing
Navigation
13 Jun
BM
View All Users
13 Jun
Add another card



Repositories 5 | Packages | People 8 | Teams | Projects | Settings

Green-core

Find a repository... | Type: All | Language: All | Customize pins | New

Documents Private

0 0 0 0 Updated 16 hours ago

Web-App Private

JavaScript 0 0 0 0 Updated 23 hours ago

Mobile-App-Backend Private

Green core Mobile App backend

JavaScript 0 0 0 0 Updated yesterday

Top languages

JavaScript

People 8 >

Thank You

Any Questions ?