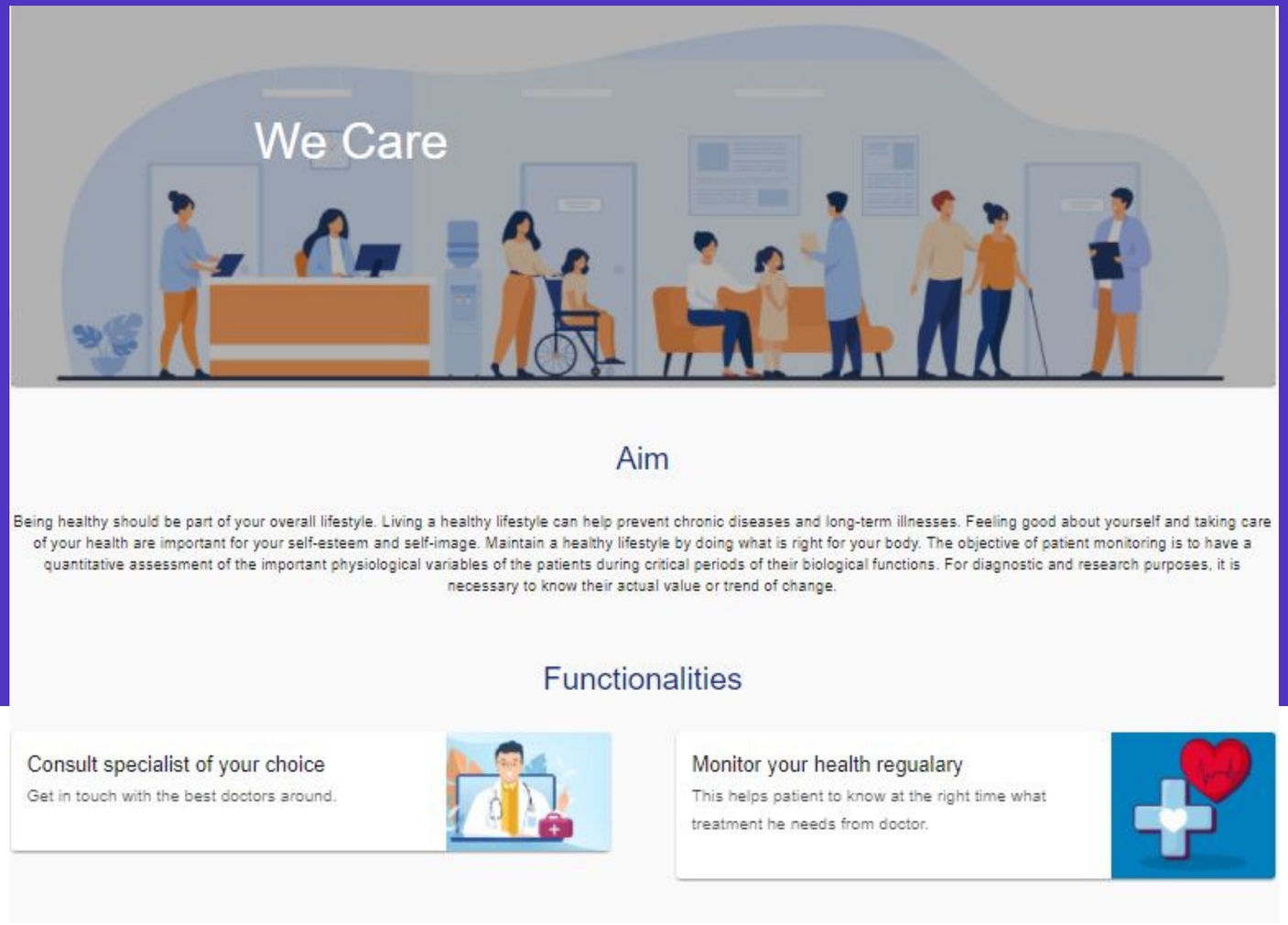
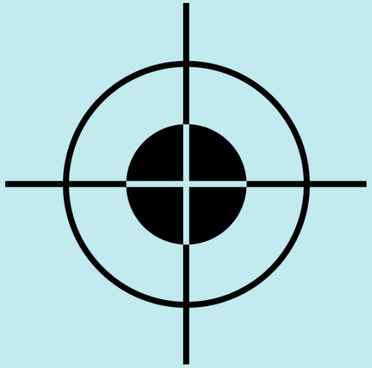


HEALTHCARE MANAGEMENT SYSTEM

Health +

Group 20
Mentor: Nitish Yadav



BJECTIVES

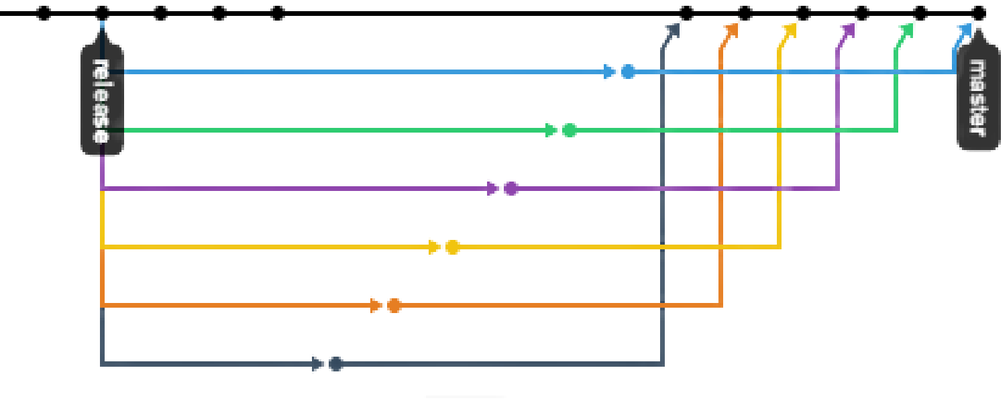
01 To understand how an application is engineered over the SDLC by following pre-existing methodologies.

02 To clearly evaluate and generate the necessary requirements and assesments

03 Detailed design for the various functionalities provided to the user

04 To develop a system that can support continuous monitoring of patient's data, notify doctors in cases of anomalies, maintain patient's medical reports, and parameters. It should also include scheduling of appointments and notices for future appointments.

Milestones



- Sprint-1:
  - Implementation of Login and Sign Up for Doctors and Patients.
  - Patient should be able to book an appointment.
- Sprint-2:
  - Implementation of function to accept or reject an appointment by the doctor.
  - Functionality to add medical prescriptions for patient.
  - Implementation of functionality to add medical parameters of patient.
- Sprint 3:
  - Implementation of charts/diagrams for different parameter of patients
  - To notify doctors through mails in case there are abnormalities in any of the parameters.

Lessons Learned

- Always Start Small then Extend
- Design and evaluate before Development
- Commit the small changes first and keep your code updated.
- Always other use branches rather than the main branch.
- Follow the process model till the end.
- Always keep buffers, it will takes longer than you expect.
- No assumptions on how much the team member knows.
- Diagrams helps to ease the implementation of functionalities.

- Knowledge of collabration platform is equally important as the frameworks.

Achievements

- Managed to use Agile Scrum Artifacts through out the project. Successfully documented all user requirements and related documents.
- Successfully built Clean, Concise and Responsive User Interface using React, HTML and JS.
- Learnt to implement backend using NodeJS and connect it with MondoDB database.
- Learnt how to manage a project by using GitHub (repositories, commits, pull requests, issues).

Shortcoming

- Less knowledge of GitHub, not maintaining the right structure of the project at the start.
- Fewer efforts in UI design.
- The development started late because the Initial focus was on learning new technologies and tools, which took greater time than expected.
- It took us a long time for merging the front-end with the back-end than expected .

Project Ratings

Rating of Software Artifacts

- Code 4.4/5
- Documentation 4.5/5
- Project management 4.5/5

Overall Project Rating

- 8.93/10

Why Agile?

Iterative approach to project management

Helps teams deliver value to their customers faster and with fewer headaches

Requirements, plans, and results are evaluated continuously so teams have a natural mechanism for responding to change quickly

Distributed Scrum

Requirement Elicitation Techniques

A From the problem statement, we identified our user base and then went ahead to identify our requirements with the help of the following methods

01 Interviews

02 Survey Forms

03 Brainstorming

3 Artifacts of Scrums

Artifact is something like a tool that we make to solve a problem

PRODUCT BACKLOG

- list of work that needs to be done.
- Basically "To Do" list for the team

SPRINT BACKLOG

- User stories, bugs that need to be fixed in the current sprint cycle.
- Used MoSCoW prioritization technique

SPRINT GOAL

- Usable end product from a sprint.

Contribution

- **DEVYANI PANCHAL**
  - Scrum Master, Organizing Scrum Events and product backlogs, Scrum Planning, Ensuring of best practices of spring reviews and scrum artifacts
  - Front End: Patient Dashboard, Home Page, Medicines, Reminders, Add data, Sign In, Sign Up
  - documentation: Product backlog Estimation, MoMs, PPT
- **KIRTAN DELWADIA**
  - Front End: Doctor and Patient Dashboard, Sign In, Sign Up, Add and View Data, Appointments and Medicines
  - Demo Video, Back End and Front End Integration
  - Database Models: Appointment, Patient, Doctor, Medicine
- **SMIT MANGUKIYA**
  - MongoDb Atlas and SetUp, Express Server, Design database for application, MongoDB models, REST API endpoints on server, Front End and Back End integration, Deployed wesite on Vercel and Back End on Heroku
- **RISHABH Rathod**
  - Documentation, Forms, Doctor’s List

Contribution

- **JASAPARA DHRUMIT**
  - Activity Diagrams, Class Diagrams, Sequence Diagrams
- **PARIKH ACHAL**
  - Activity Diagrams, Class Diagrams, Sequence Diagrams
- **SAKETH RAM**
  - PPT, Development
- **KOTAK PAVAN**
  - Documentation, Requirement Elicitation, User Stories (Prioritization + Acceptance Condition), Patient Report Forms
- **PATEL AYUSHI**
  - queries for adding and fetching different parameters from the database(Documentation), Elicitation techniques, Identifying and prioritizing user stories for different sprints, Sprint backlog and sprint log for different sprints
- **HARDIK MADHWANI**
  - Backend Queries for adding and fetching the parameters, sending mails
  - Documentation: Unit Testing, Sequence and activity Diagrams

- **Risk Assessment**