

Project Development Phase

Sprint 4

Date	14 November 2022
Team ID	PNT2022TMID41539
Project Name	Project-Signs with Smart Connectivity for Better Road Safety
Maximum marks	8 Marks

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-4	Develop A Web Application Using Node-RED Service.	USN-8	Develop The Web Application Using Node-RED Configure the Node-RED flow to send data to the IBM IoT platform.	5	High	1.Mugila R 2.Ishwariya P 3.Kalpana T 4.Shreein Fathima S

Node-Red flow created

The screenshot displays the Node-RED web interface in a browser. The address bar shows the URL: `node-red-xqkno-2022-10-03.mybluemix.net/red/#flow/895a58d22de00831`. The interface includes a left sidebar with node categories (common, function), a central workspace for 'Flow 1', and a right sidebar with dashboard controls.

Flow 1 Diagram:

- Input:** A blue **IBM IoT** node (connected) feeds into a **Batch 11** node.
- Processing:** The flow branches into five parallel paths, each consisting of a function node followed by a message node:
 - temperature:** `msg.payload` function node connected to a `temperature` message node.
 - MESSAGE:** `MESSAGE` function node connected to a `Message` message node.
 - sign:** `sign` function node connected to a `Sign` message node.
 - Speed:** `Speed` function node connected to a `Speed` message node.
 - Visibility:** `Visibility` function node connected to a `Visibility` message node.

Right Sidebar (Dashboard):

- Layout:** Site, Theme
- Tabs & Links:**
 - Home
 - Signs with smart connectiv
 - Hall AC
 - Smart switch board

The Windows taskbar at the bottom shows the time as 05:55 on 19-11-2022, with icons for various applications and system status.

Device connected

The screenshot displays the Node-RED web interface in a browser. The address bar shows the URL: `node-red-xqkno-2022-10-03.mybluemix.net/red/#flow/895a58d22de00831`. The interface is divided into several sections:

- Left Panel (Nodes):** Contains a search bar and two categories of nodes: **common** (inject, debug, complete, catch, status, link in, link call, link out, comment) and **function** (function, switch).
- Flow Canvas:** Shows a flow named "Flow 1" with a "Batch 11" node connected to an "IBM IoT" node. The "IBM IoT" node has a green "connected" status indicator. It is connected to several function nodes on the right.
- Right Panel (Edit ibmiot in node):** A configuration panel for the selected "IBM IoT" node. It includes a "Delete" button, "Cancel", and "Done" buttons. The "Properties" section lists the following settings:
 - Authentication:** API Key
 - API Key:** IBMIOT APIKEY
 - Input Type:** Device Event
 - Device Type:** ☒ All or +
 - Device Id:** ☐ All or device id e.g. ab12cd231a21
 - Event:** ☒ All or +
 - Format:** ☐ All or json
 - QoS:** 0
 - Name:** IBM IoT
 - Service:** registered
 - Enabled:** ☐ Enabled
- Dashboard Panel:** Located on the far right, it shows a "dashboard" tab with sections for "Layout", "Site", and "Theme". The "Tabs & Links" section lists:
 - Home
 - Signs with smart connectiv
 - Hall AC
 - Smart switch board

The Windows taskbar at the bottom shows the time as 23:57 on 18-11-2022, along with various system icons and application shortcuts.

Verify your identity - m x Signs with smart conne x IBM Service Details - IBM Cl x IBM Watson IoT Platform x Node-RED : node-red-x

node-red-xqkno-2022-10-03.mybluemix.net/red/#flow/895a58d22de00831

Node-RED

filter nodes

Flow 1

Batch 11

common

- inject
- debug
- complete
- catch
- status
- link in
- link call
- link out
- comment

function

- function
- switch

IBM IoT

connected

msg

ter

M

sig

Sp

Vis

Edit debug node

Delete Cancel Done

Properties

Output msg.payload

To ☒ debug window ☐ system console ☐ node status (32 characters)

Name Name

Enabled

dashboard

Layout Site Theme

Tabs & Links

- Home
 - Signs with smart connectiv
 - Hall AC
 - Smart switch board

05:56 19-11-2022

Node-RED output

The screenshot displays the Node-RED web interface in a browser. The address bar shows the URL: `node-red-xqkno-2022-10-03.mybluemix.net/red/#flow/895a58d22de00831`. The interface includes a left sidebar with node categories (common, function), a central workspace for 'Flow 1', and a right sidebar for the debug console.

Flow 1 Diagram:

- Input:** A `Batch 11` node feeds into an `IBM IoT` node (status: connected).
- Processing:** The `IBM IoT` node has five outgoing connections to function nodes:
 - `msg.payload`
 - `temperature`
 - `MESSAGE`
 - `sign`
 - `Speed`
 - `Visibility`
- Output:** Each function node is connected to a corresponding output node:
 - `temperature` connects to a `temperature` output node.
 - `MESSAGE` connects to a `Message` output node.
 - `sign` connects to a `Sign` output node.
 - `Speed` connects to a `Speed` output node.
 - `Visibility` connects to a `Visibility` output node.

Debug Console Output:

The debug console shows four messages received at 11/19/2022, 6:00:19 AM, 6:00:24 AM, 6:00:30 AM, and 6:00:35 AM. Each message is a JSON object with the following structure:

```
{
  "Temperature": 296.14,
  "Message": "",
  "Sign": "",
  "Speed": "SPEED MESSAGE: SLOW DOWN, spee...",
  "Visibility": "VISIBILITY: Clear Weather"
}
```

The bottom of the screen shows a Windows taskbar with various application icons and a system clock indicating 06:01 on 19-11-2022.

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-4	Develop A Web Application Using Node-RED Service.	USN-9	Use Dashboard Nodes For Creating UI(Web App) Create use dashboard nodes to visualize the data in graphical format.	5	High	1.Mugila R 2.Ishwariya P 3.Kalpna T 4.Shreein Fathima S

Nodes created

The screenshot displays the Node-RED Dashboard layout editor interface. The main workspace is titled "Dashboard layout editor : Home" and contains three panels, each with a width of 6:

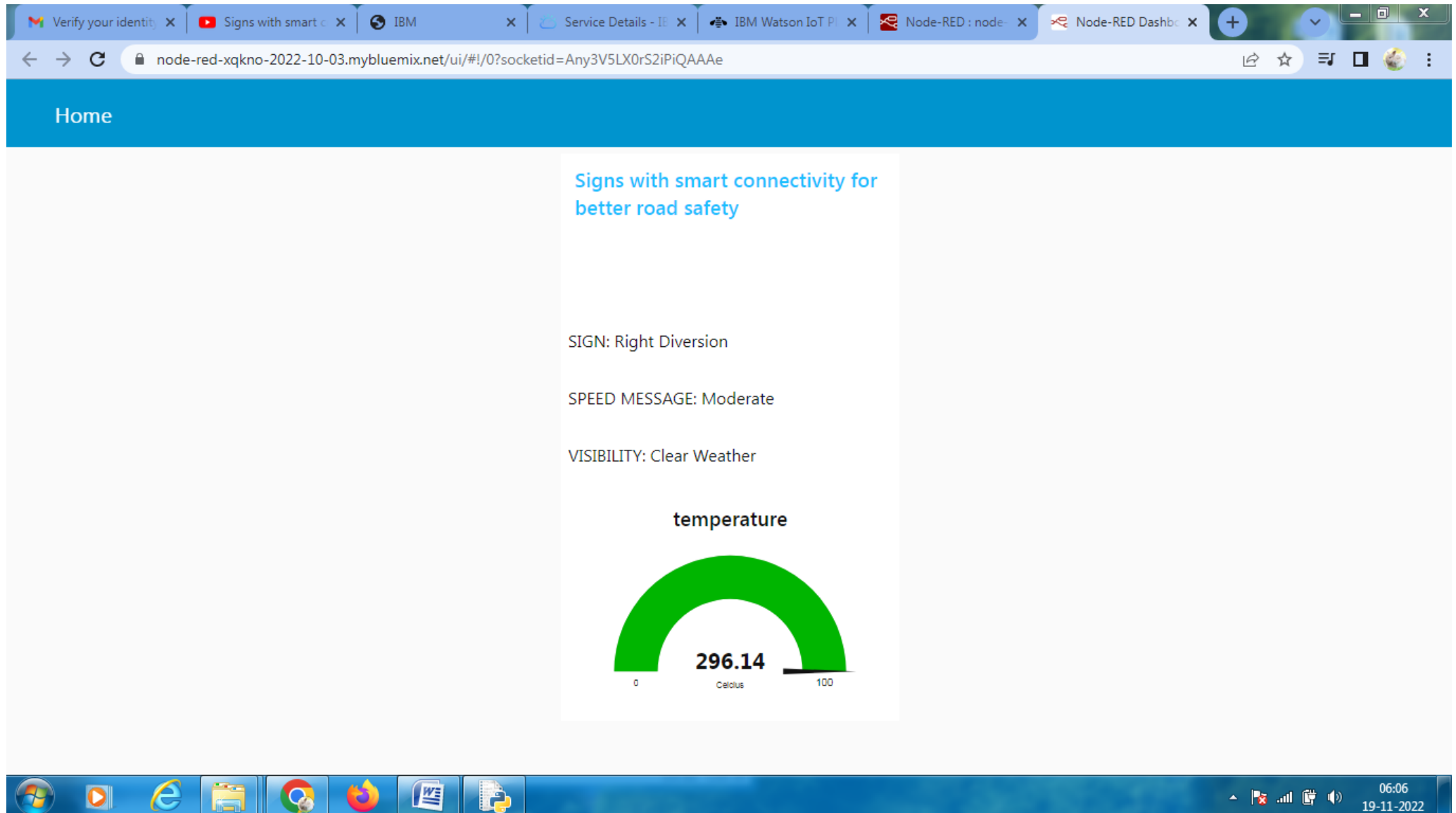
- Signs with smart connectivity for better road safe:** A 5x5 grid containing five templates: "Message template", "Sign template", "Speed template", "Visibility template", and "temperature gauge".
- Hall AC:** An empty 6x1 grid.
- Smart switch board:** An empty 6x1 grid.

The right sidebar shows the dashboard structure with tabs and links:

- Layout:** The active tab.
- Site:** A tab for site configuration.
- Theme:** A tab for theme configuration.
- Tabs & Links:** A list of dashboard elements:
 - Home
 - Signs with smart connectivity
 - spacer 6x1
 - Message
 - Sign
 - Speed
 - Visibility
 - temperature
 - Hall AC
 - Smart switch board

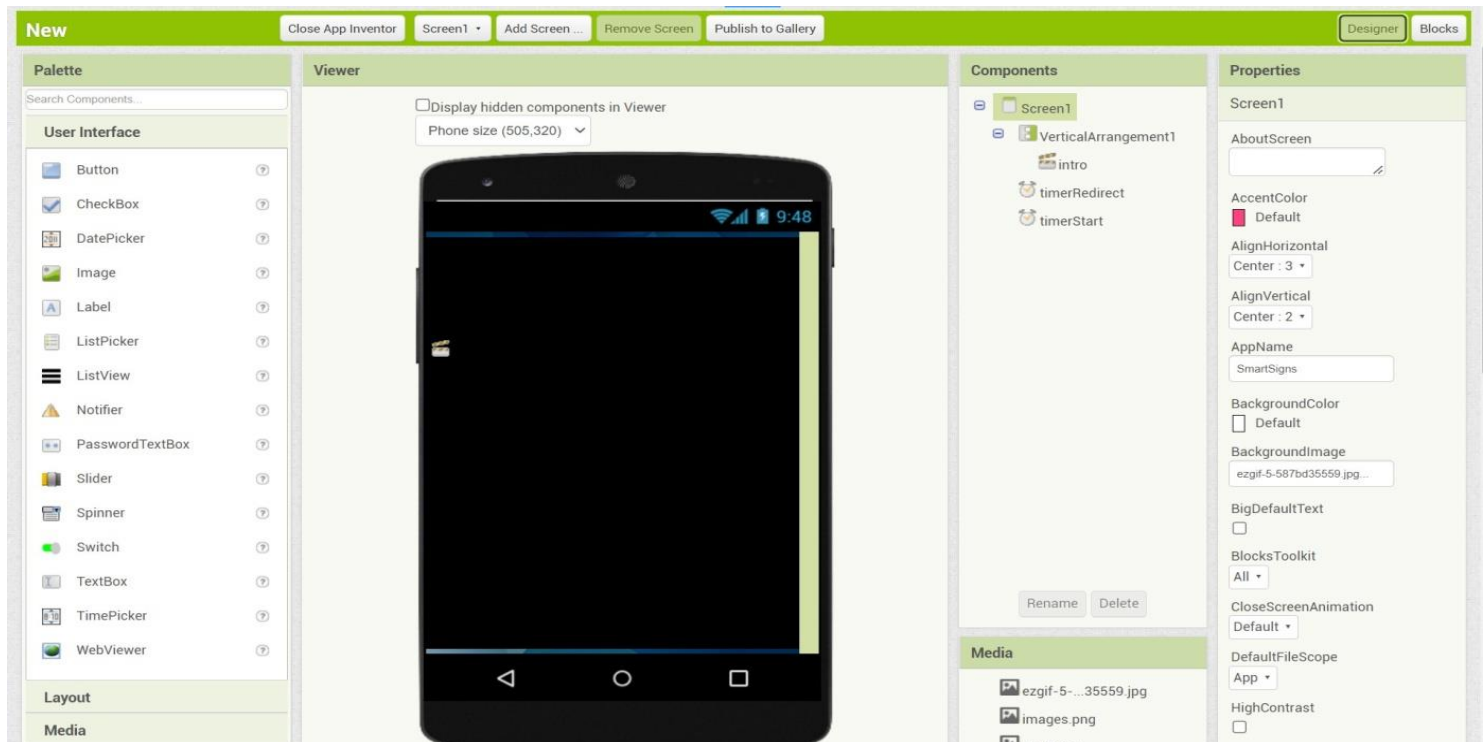
The bottom of the image shows a Windows taskbar with various application icons and a system clock indicating 06:06 on 19-11-2022.

Node Output



Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-4	Develop A Mobile Application Using MIT App inventor.	USN-10	Use MIT App For Creating Mobile application Build the app. Using MIT AI2 companion(in mobile) by connecting it with the app builded(click connect in app inventor then click AI companion and scan the QR code in mobile companion- it will be connected) we can see the Road Safety informations in mobile.	10	High	1.Mugila R 2.Ishwariya P 3.Kalpana T 4.Shreein Fathima S

MIT app inventor – Screen 1 Frontend



Screen 1 – Backend

The screenshot displays the App Inventor interface for creating a new screen. The top bar is green and contains the text 'New' on the left, and buttons for 'Close App Inventor', 'Screen1', 'Add Screen ...', 'Remove Screen', and 'Publish to Gallery' in the center. On the right of the top bar are buttons for 'Designer' and 'Blocks'.

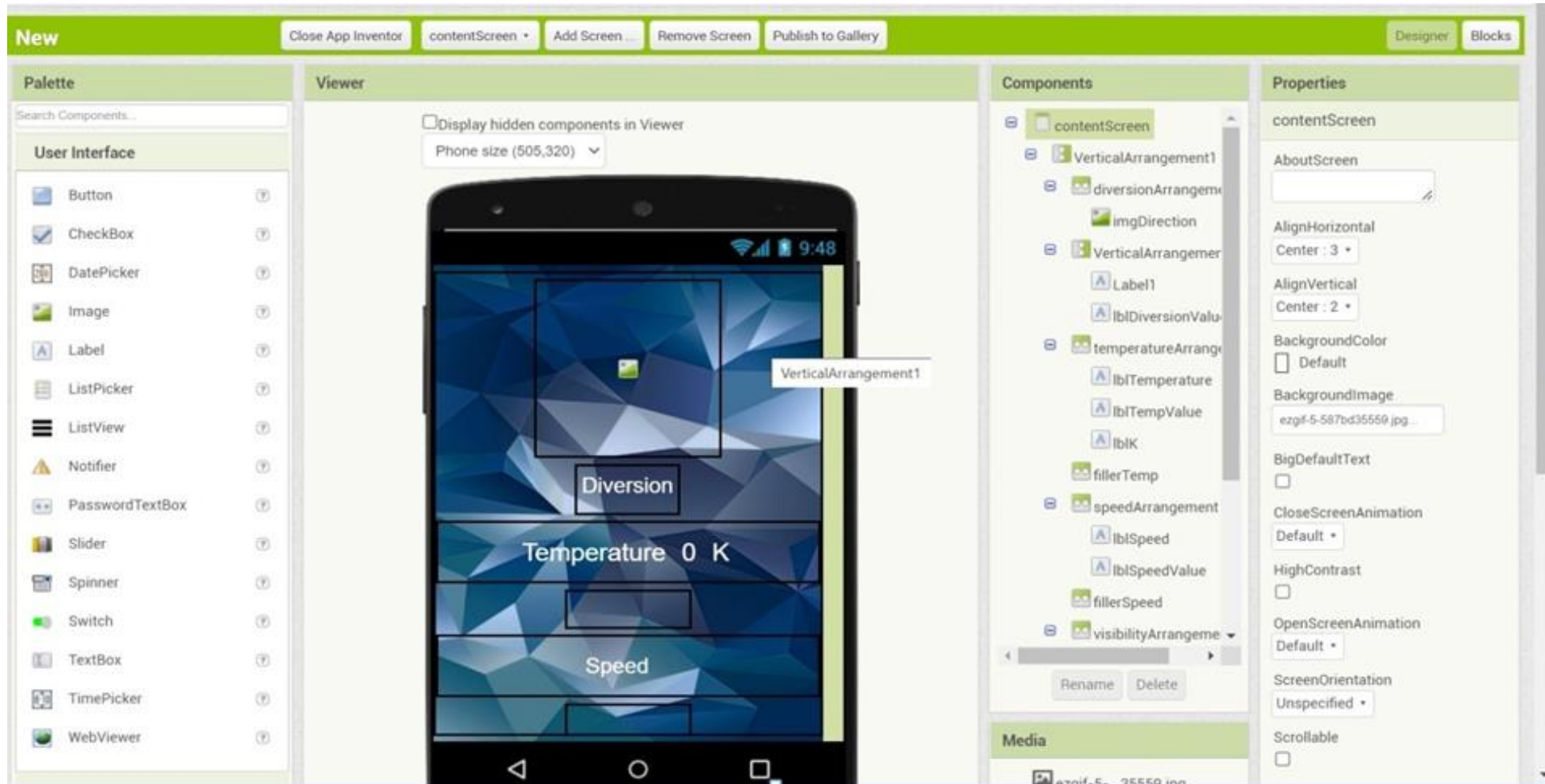
The left sidebar is divided into two main sections: 'Blocks' and 'Media'. The 'Blocks' section is further divided into 'Built-in' and 'Screen1'. Under 'Built-in', there are categories: Control, Logic, Math, Text, Lists, Dictionaries, Colors, Variables, and Procedures. Under 'Screen1', there is a 'VerticalArrangement1' component which contains three sub-components: 'intro', 'timerRedirect', and 'timerStart'. At the bottom of the 'Blocks' sidebar are 'Rename' and 'Delete' buttons. The 'Media' section is currently empty.

The main 'Viewer' area shows the visual representation of the screen. It contains two event-driven blocks:

- A 'when timerStart .Timer' block with a 'do' block containing 'call intro .Start'.
- A 'when timerRedirect .Timer' block with a 'do' block containing 'open another screen screenName LoginPage'.

At the bottom left of the viewer, there are two warning icons (a yellow triangle and a red X) both with a '0' next to them, and a 'Show Warnings' button below them. On the right side of the viewer, there is a blue backpack icon at the top, and a vertical toolbar with icons for zooming in (+), zooming out (-), and a trash can at the bottom.

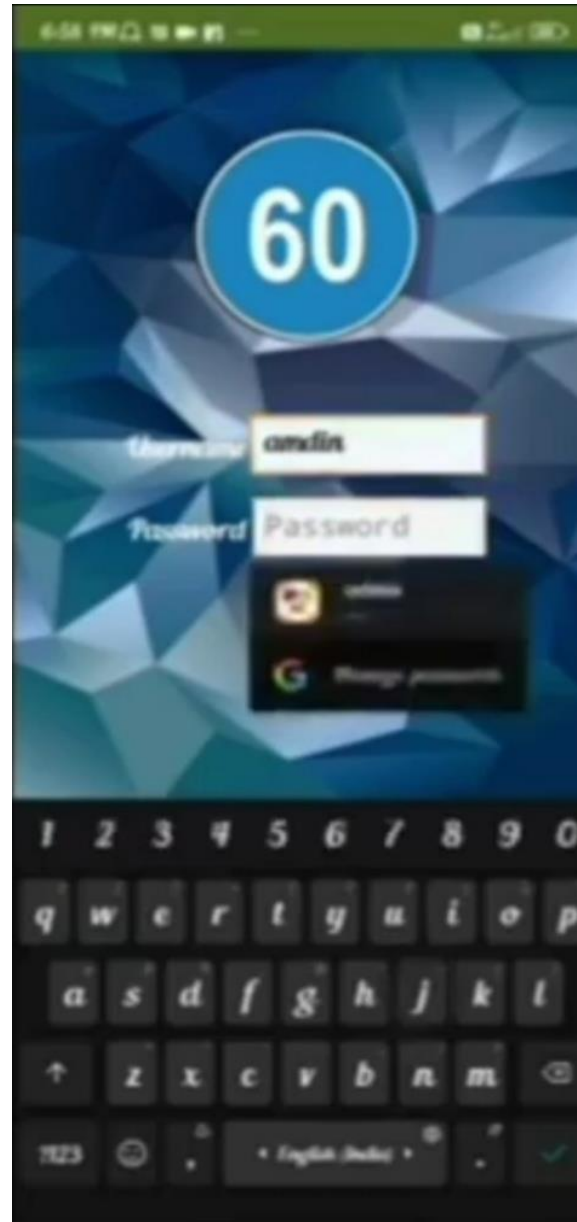
Screen 2 – Frontend



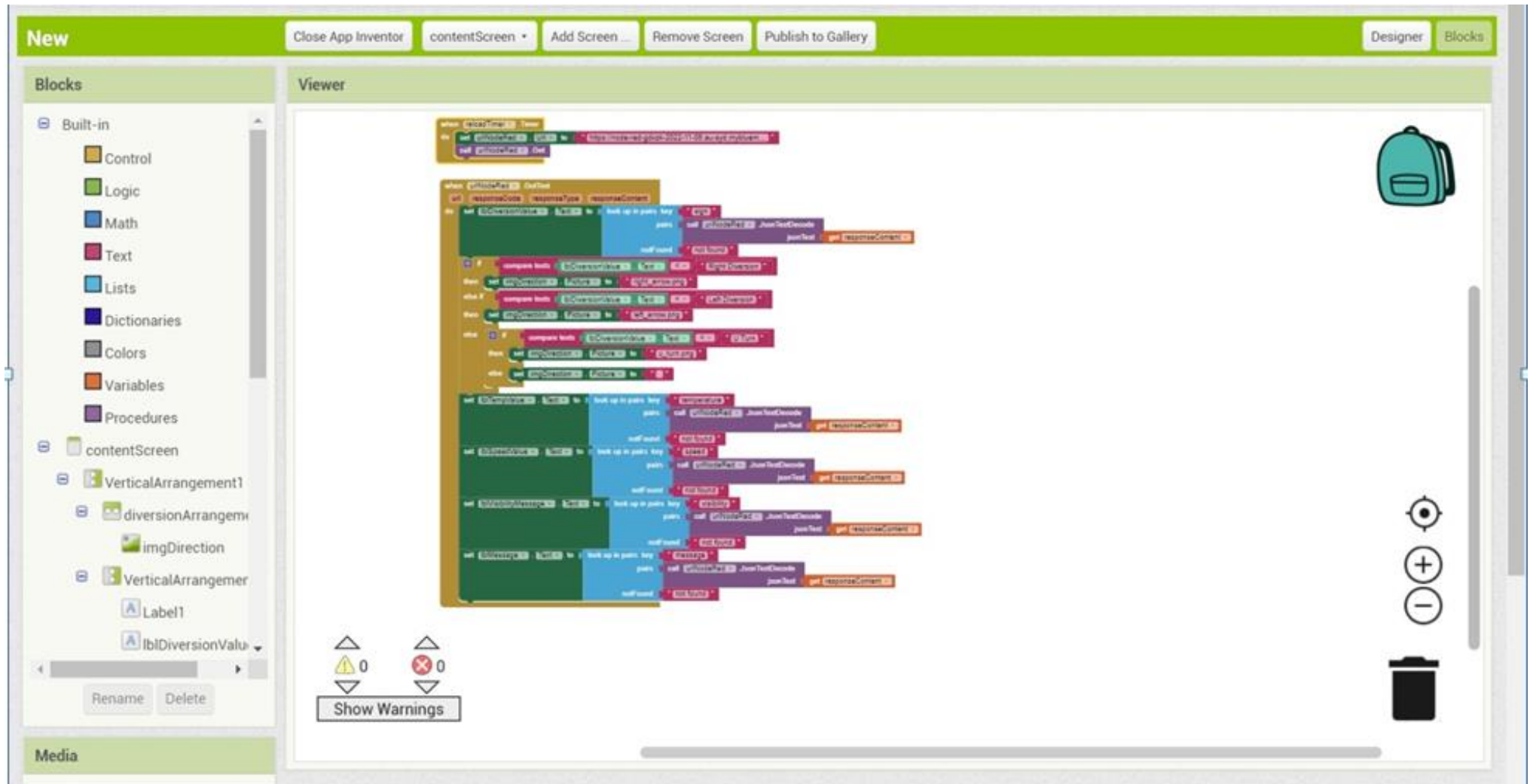
Screen 2 – Backend



Screen 2- Frontend (Result after connecting MIT AI2 mobile companion)



Screen 3 – Backend



Screen 3 – Frontend (Result after connecting MIT AI2 mobile companion)



Hence the project is done.