

**PROGRESSIVE WEB APP REPORT**

on

***PIZZA WEBSITE***

Submitted by

**UJJAWAL JAISWAL**

**Registration No: 11411752**

**Programme Name: B.TECH (CSE)**

Under the Guidance of

**Mr. Roshan Srivastava**

**School of Computer Science & Engineering**

**Lovely Professional University, Phagwara**

(August-November, 2017)

*PIZZA ORDER WEBSITE*

*A Progressive Web App is:*

****Progressive**** - Works for every user, regardless of browser choice because it's built with progressive enhancement as a core tenet.

****Responsive**** - Fits any form factor: desktop, mobile, tablet, or whatever is next.

****Connectivity independent**** - Enhanced with service workers to work offline or on low-quality networks.

****App-like**** - Feels like an app, because the app shell model separates the application *functionality* from application *content* .

****Fresh**** - Always up-to-date thanks to the [service worker](https://developers.google.com/web/fundamentals/getting-started/primers/service-workers) update process.

****Safe**** - Served via HTTPS to prevent snooping and to ensure content hasn't been tampered with.

****Discoverable**** - Is identifiable as an "application" thanks to [W3C manifest](https://developers.google.com/web/updates/2014/11/Support-for-installable-web-apps-with-webapp-manifest-in-chrome-38-for-Android) and [service worker registration](https://developers.google.com/web/fundamentals/instant-and-offline/service-worker/registration)scope, allowing search engines to find it.

****Re-engageable**** - Makes re-engagement easy through features like push notifications.

****Installable**** - Allows users to add apps they find most useful to their home screen without the hassle of an app store.

****Linkable**** - Easily share the application via URL, does not require complex installation.

Steps to make our website progressive:-

1- Make a website which we want to make progressive.

2- Install and verify web server.

### 3- Register the service worker in the app.js file.

### 4- Use service workers to cache the forecast data.

### 5- Intercept the network request and cache the response.

6- Add the manifest file to our website and register in index.html.

*The Service Worker's Life Cycle goes thus:*

****Install**** : An install event is triggered the first time a user visits the page. During this phase, the service worker is installed in the browser. During this installation, you can cache all the static assets in your web app.

The filesToCache variable represents an array of all the files you want to cache.

The cacheName refers to the name given to the cache store

****Activate****: This event is fired when the service worker starts up.

****Fetch****: This event helps serve the app shell from the cache. caches.match()dissects the web request that triggered the event, and checks to see if it's available in the cache. It then either responds with the cached version, or uses fetch to get a copy from the network. The response is returned to the web page with e.respondWith().