

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Khetdan Phopat Samuel Olsson Amjad Alakrami |  | Jokim Flink  Andreas Jillram  Daniel Åkerlund  ABB industrigymnasium |  | 15/12/2018 |

W.M.C

Jobba mer bekvämt

Innehållsförteckning

1. [1 Inledning 2](#_Toc532855927)

[1.1 Syftet](#_Toc532855928)

1. [2 Funktionsbeskrivning och Metod 3](#_Toc532855929)

[2.1 Appen Funktionsbeskrivning](#_Toc532855930)

[2.2 Armatur Funktionsbeskrivning och Metod](#_Toc532855931)

1. [3 Material 5](#_Toc532855932)
2. [4 Resultat 6](#_Toc532855933)
3. [5 Detaljbeskrivning 7](#_Toc532855934)

[5.1 Armaturens Detaljbeskrivning](#_Toc532855935)

[5.2 Appens Detaljbeskrivning](#_Toc532855936)

[5.3 GoogleHome Detaljbeskrivning](#_Toc532855937)

1. [6 Kod Utdraga 10](#_Toc532855938)
2. [7 Utvärdering 11](#_Toc532855939)

[7.1 Positiva sidor](#_Toc532855940)

[7.2 Utvecklingsmöjligheter](#_Toc532855941)

[7.3 Lampans framtid](#_Toc532855942)

# 1 Inledning

W.M.C är en unik kontorslampa som har olika funktioner som gör att arbetet bakom kontorsbordet blir mer bekvämt för ögonen, vilket i sin tur leder till att man orkar jobba under en längre tid utan att ens ögon blir störda, och speciellt när man jobbar bakom datorskärmen, skriver med papper och penna, eller läser en bok.

Namnet W.M.C är en förkortning för (Work More Comfortable), och den kom från själv målet med våran ide, vilket är att gör det mer bekvämt att jobba bakom kontoret, i båda skolan och i vanliga arbetslivet. Iden kom till våra hjärnor när vi märkte att lamporna vi har över våran skolbänk har inga andra funktioner än att bara lysa, och då ville vi hitta på en bättre lamp ide som är mer funktionellt än de vanliga lampor som vi ofta ser överallt. Sedan tänkte vi på vilka funktioner som är viktiga för en kontorslampa? Och hur kan vi få de funktionerna att funka? I en kontorslampa så är det viktigt att det är flexibelt, alltså till exempel att kunna sänka o höja den, att kunna kontrollera om ljuset sak spridas över hela bordet eller om man vill ha det över ett specifikt ställe, och det vi tyckte var viktigast av allt att ha i en kontorslampa är att det måste vara mer enkelt att kontroller om det ska vara tänd eller släkt, om den ska lysa kall eller varm ljus. Och de funktionerna kan man ha i en vanlig fysisk knapp men den ide strå mot vårt syfte med en funktionell lampa, vilket fick oss att tänka på att programmera en mobil app som kan styra den lampa, och då kan folk tycka att ”tänk om mobilen är urladda, hur ska man kunna kontrollera lampan då?” ju, det är helt rätt tänkt OM en enhet som GoogleHome inte exciterade. Med hjälp av GoogleHome kan man styra över lampa bara genom att säga vad man att lampan ska göra, och då är det inte så stort problem om mobilen är urladdad.

## 1.1 Syftet

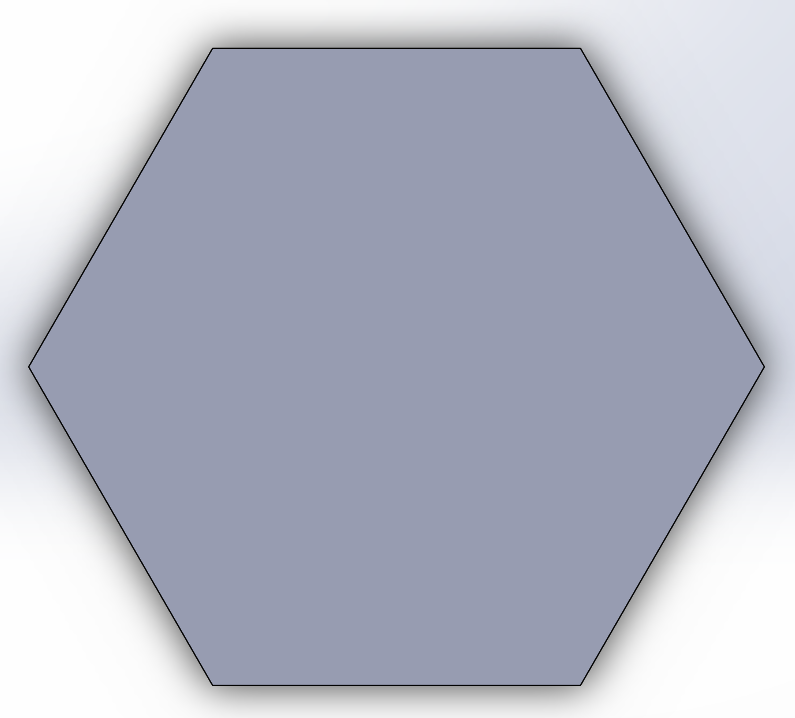
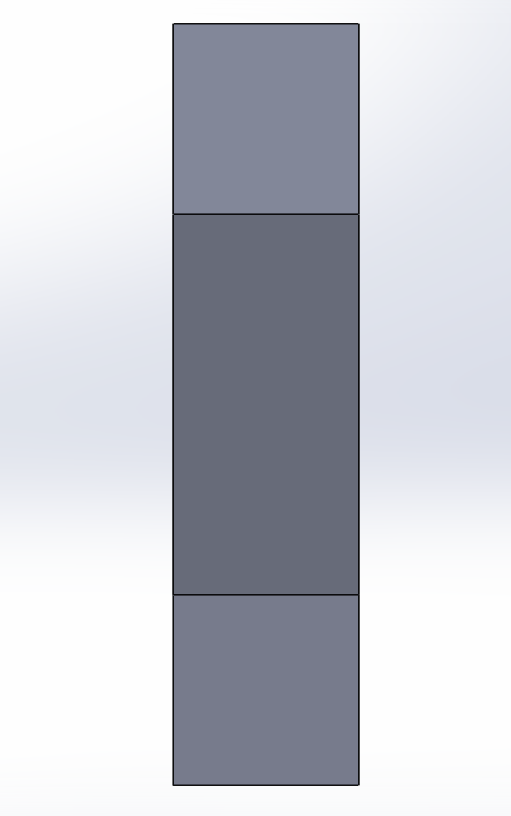
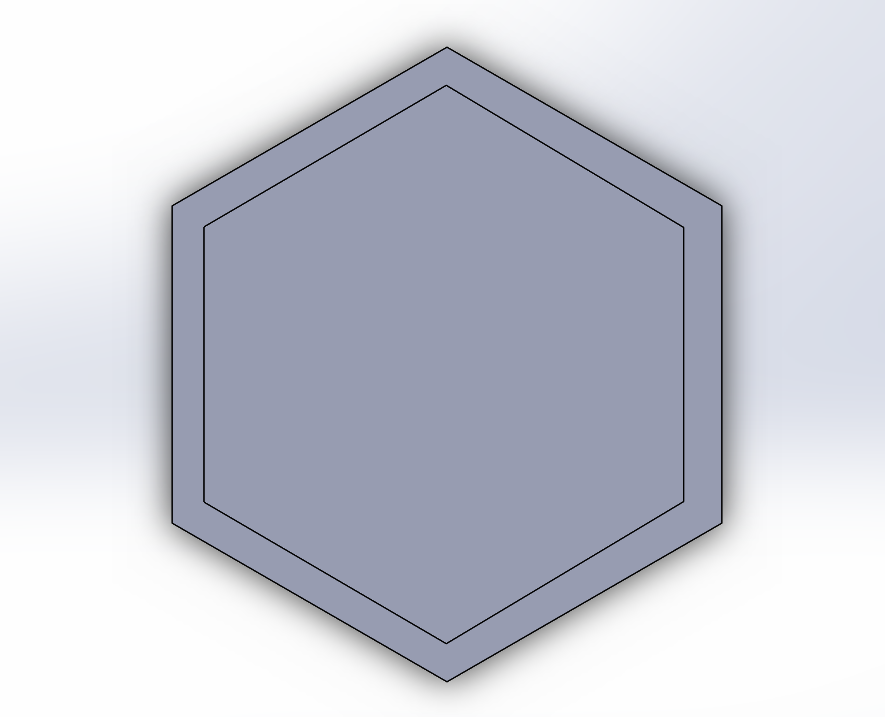
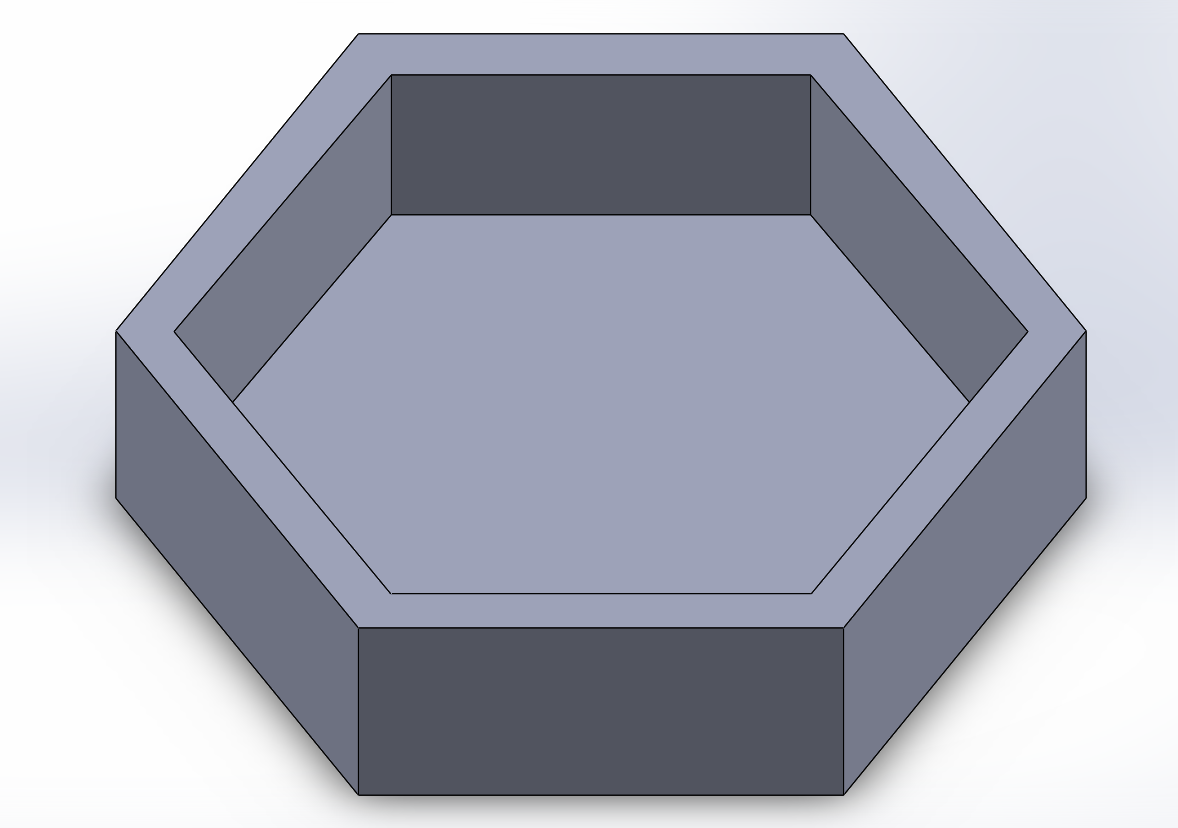
Syftet med vår smarta lampa är att gör det mer bekvämt att till de som jobbar bakom kontoret, eller till eleverna som sitter o studera bakom skolbänkar, speciellt i de skolor där dator tar en stor plats av eleverna studier.

# 2 Funktionsbeskrivning och Metod

## 2.1 Appen Funktionsbeskrivning

Eftersom vi inte vill ha några fysiska knappar som tänder lampan, så blir Appen- den bästa ersättare. Appen består av två olika skärmar. Den första skärmen så finns det en stor knap som har funktionen att tända lampan, och när man trycker på den så tänds lampan samtidigt som andra skärmen kommer fram. I andra skärmen så finns det två ”Range slider” en har funktionen att kontrollera styrkan på de kalla ledstrippar, och den andra kan kontrollera styrkan på de varma.   
I andra skärmen så finns det också en liten knapp som har funktionen släcka lampan och gå tillbaka till den första skärmen.

## 2.2 Armatur Funktionsbeskrivning och Metod

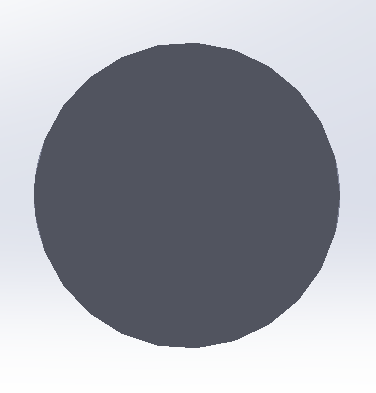
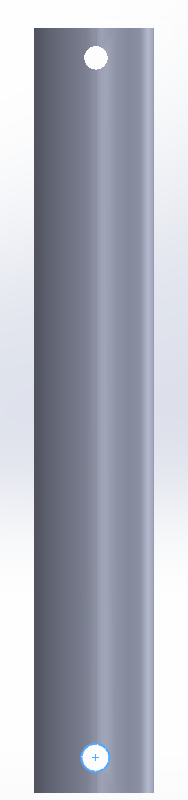
När vi först kom på iden så såg armaturen ut som en hexagon med sex rörliga väggar runt om basplattan, alltså väggarna ska vara ihopsatta med basplattan med hjälp av gångjärn så att de kan rotera i förhållande till basplattan. Och 5cm neråt från basplattan så kommer det sitta en till platta som dämpar ljus så den inte blir för stark och obehaglig, orsaken till at vi valde just 5cm var att efter olika tester med olika avstånd så var 5cm det rätta avståndet som ger tillräckligt stark och behaglig ljus. (I skissen så syns inte gångjärnet och plattan som dämpar ljuset, utan det är bara en skiss för själva utseendet).

Basplattan Sid vy Ovan vy Ovanvinkel

Men efter några diskussioner så kring vårt första val av armaturens utseende, så märkte vi att den kommer bli onödigt mycket användning av material som inte spelar någon roll med lampans funktion, vilket betyder att kostnaden på lampan kommer vara dyrare utan någon viktig syfte. En till hinder med mycket materialanvändning är att det kommer vara miljöskadlig, eftersom de materialen vi hade till gång till vara någon sortplast, och kortfattat så används råoljan ”en av fossila resurser” som råmaterial I de flesta plaster, vilket innebär mycket koldioxidutsläpp som skadar miljön. En av de viktigaste hindrar med att vår armatur ska se ut som en hexagon var att det skulle bli mycket krångel med rörliga väggar, eftersom det är ganska svårt att få sex väggar att röra sig lika mycket samtidigt. Lösningen på dessa hindrar var så enkel som att minska antalet sidor på basplattan, utan att ändra på vårt mått, då kommer vi få mindre area, vilket betyder mindre material som behövs, och då kommer vi få mindre antal väggar att kontrollera. Och figuren vi bestämde var en kvadrat, eftersom om vi skulle ha valt en pentagon så skulle de inte göra så mycket skillnad, och om vi skulle ha valt en triangel så skulle armaturen ha blivit otillräckligt stor, så en kvadrat kändes som den bästa valet.

# 

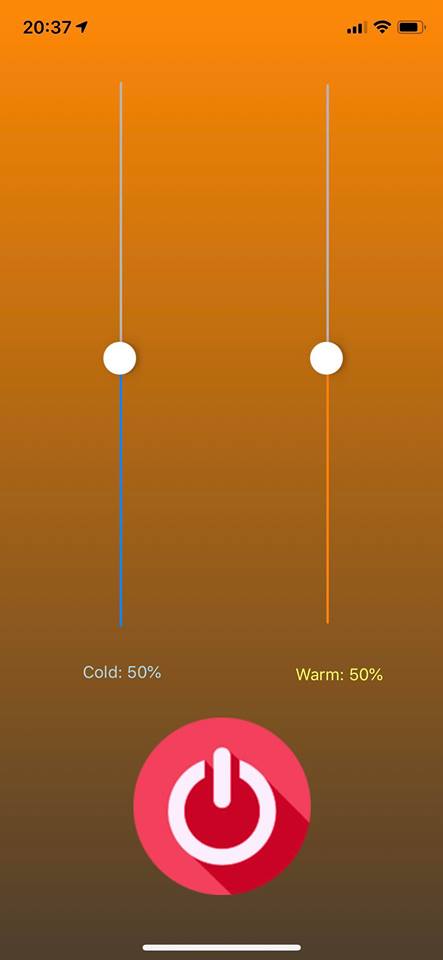
# 3 Material

Vår armatur ät som sagt uppbyggd av två sorters plast, plexiglas som kallas också för akrylplast. ”Akryl eller PMMA används ofta som ett alternativ till glas. Akryl är en termoplast med ca 92 % transparens. Plasten är väder- och åldersbeständig och har många användbara egenskaper, är enkel att hantera och bearbeta. Den låga kostnaden gör den till ett självklart val för många användningsområden”[[1]](#footnote-2). Och plasten som användes i 3D skrivaren, eftersom den är ganska lätt och hållbar. Plexiglaset har vi använt som ljus dämpare platten, anledningen till att vi valde just det materialet är för att den gör att ljuset blir snällare för ögat utan att göra ljuset så mycket svagare. Den 3D skrivna delar som spelar störst roll i armaturen är de delar som sätter fast basplattan och plexiglaset ihop med 5cm avstånd emellan. Och de delarna är fyra likadana pinnar som har ett håll i varje ända, och i varje sådan håll finns in liten pinne som sticker ut från båda sidorna.

Fram vy Under vy

Men armaturen är mest uppbyggd av cellpannå, kallad även kapaskiva. Båda basplattan och väggarna är byggda av detta materialet, och anledningen till att vi valde just den matrealet är för att den är båda lätt och ganska hållbar. Cellpannå finns i olika färger, men vi valde färgen vit, eftersom den inte absorberar så mycket ljus, vilket leder till att inte så mycket ljus som försvinner.

# 4 Resultat

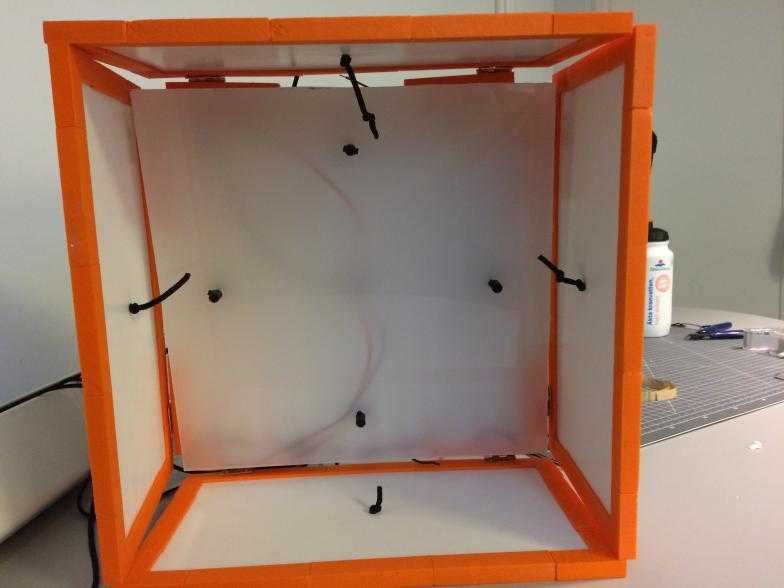
* Appen

Första skärmen Andra skärmen

* Armaturen

## 

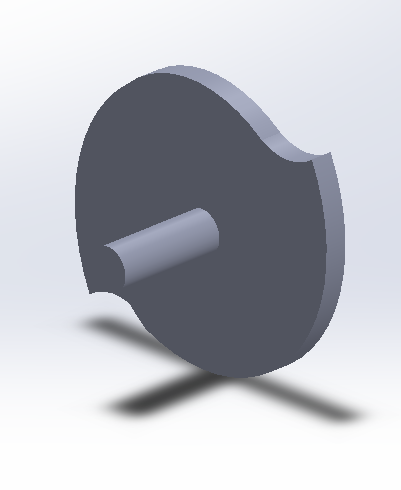
Projektor Väggen öppet Väggen Stängd



Armaturen under ifrån

# 5 Detaljbeskrivning

## 5.1 Armaturens Detaljbeskrivning

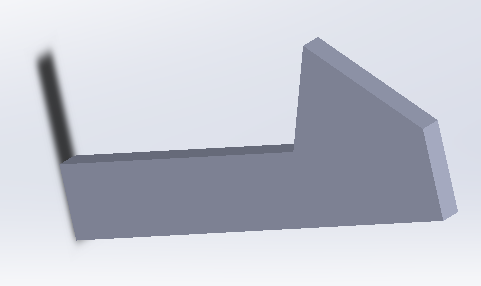
De funktionerna vi tyckte var viktigast i en lamp armatur var att kunna höja och sänka den, och att kunna styra ljusspridning, alltså om man vill att ljuset ska vara bara över ett specifikt ställe eller om den ska spridas över hela rummet. Det är oftast de funktionerna som en kontorslampa behöver.   
Vi tänkte att dessa två funktioner ska var ihop kopplade, alltså när man drar ner lampan så betyder det att man vill ha ljuset över ett specifikt ställe, då stäng väggarna så att det inte blir så mycket ljus spridning, och när man höjer upp så öppnas väggarna så att ljuset sprids över hela rummet.   
De funktionerna liknar precis en projektorduk. Alltså en fjäder som sitter i en tub, och den är bunden med tuben från båda änderna. Från ena endan så är tuben stängd så att fjäder inte ska ramla ut, och andra ända så är det också stängd men den har också två andra viktiga delar. Den första är en liten pinne som sticker ut, de är kopplad med en platta som i sin tur är kopplad med fjäder in i tuben.

Kuggar

Plattan

Pinnen 

Pinnen som är kopplad till plattan.

När man vrider på den pinnen så vrids plattan också och sen spänns fjädern inuti tuben, och då måste den snurra tillbaka så den slipper vara spänd. När fjädern snurrar tillbaka så gör platten också det, och då stoppas snurrningen med hjälp av en liten del som ser ut som en krok ungefär, som krokar med kuggarna som sticker från plattan, och då kan fjädern inte snurra tillbaka. Den delen är delvis rörlig, den kan röras uppåt och tillbaka neråt. Så när fjädern blir tillräckligt spänd så snurrar den tillbaka tillräckligt fort så att den lilla kroken inte kan håll fast plattan, vilket gör att fjädern kan snurra tillbaka till den första läge den var på innan man snurrade de, alltså den blir ospänd igen.

Delen som ser ur som en krok.

Vår armatur är kopplad med den pinnen som sticker ut, med hjälp av ett rep, det repet är snurrad runt den pinnen, och den är kopplad i mitten av basplattan i armaturen, så när man drar ner lamparmaturen så snurras den pinnen som sedan spänner fjädern. om man drar ner armaturen lagom mycket så blir fjädern inte så mycket spänd så den får tillräckligt med hastighet som kan pusha kroken uppåt, då stannar lampan vid den låga platsen. Men om man drar den tillräckligt mycket neråt, då spänns fjädern så mycket så att den får hög hastighet som kan lyfta upp kroken, och då går lampan tillbaka till den höga positionen.   
Varje vägg är kopplad med ett rep till det repet som kopplar basplattan med tuben, så när lampan vid den höga positionen då är repet mellan armaturen och tuben kort, vilket innebär att repen som är kopplade med väggarna blir spända så de öppnar upp väggarna. Och när man sänker ner lampan, så får repet mellan armaturen och tuben mer längd, vilket gör att repen som är kopplade till väggarna är inte längre spända, och det gör att väggarna stängs ner.

## 5.2 Appens Detaljbeskrivning

För att få appen att kontrollera lampan så behöver man tre olika delar:

* Databas
* Backend
* Frontend
* En mikrokontroll

De informationen som behövs för lampans funktion lagras i databasen. Information som lampans namn (om den är kall eller varm), och styrkan på de lamporna.

|  |  |  |
| --- | --- | --- |
| LightName | StrWarm | StrCold |
| LightName | 500 | 500 |

Så här ser vår databas ut, och det som står här att värdet ”styrkan” på den varma lampan ”StrWarm” ska vara 500, alltså ungefär 50%, och samma sak med den kala ljus ”StrCold”.   
För att man ska kunna använda och ändra de värden så behöver vi den delen som kallas för Backend. I en Backend så skapar vi olika funktioner som tex, Get (För att hämta värden), Patch (För att ändra värden), Post (För att skicka värden) och Delete (För att ta bort värden). För att kunna använda de funktionerna så behöver man antigen en Frontend, eller en mikrokontroll.   
En Frontend är själva appen som konsumenter använder. I en Frontend så har varje knapp i speciell funktion (Tända, släcka och ändra ljusstyrka), de funktionerna motsvarar (Post eller Patch, Delete eller Patch, Patch).   
I komponent 1 (On knappen) i första skärmen i vår app, så har vi använd funktionen post, och de värdena vi ville skicka var [StrWarm, 500. StrCold, 500], alltså vi säger åt Backend att den ska skicka de värdena till databasen när vi trycker på knappen. I andra skärmen, så har vi 3 olika komponent, den första är en Off knapp, och funktionen med den är att släcka lampan. Funktionen vi programmerad för att få den fysiska funktionen (släcka lampan) är en Delete funktion. Det som händer när man trycker på den knappen är att vi säger åt Backend att den ska ta bort värden [StrWarm, 500. StrCold, 500] från databasen, och då när vi inte har något värde alls i databasen då blir lampan släckt. De två andra komponenten är Range slider, och de har sammafunktion, vilket är att ändra ljuse styrkan, och programmerings funktionen vi använde är Patch. Och vi använder en Range slider för StrCold (kala ljus) och en för StrWarm (varma ljuset). Och det som händer när man ändrar värdet i slider är att vi säger åt Backend att den ska ändra värdet i en viss kolumn (beroende på vilket slider vi ändrar värdet), till exempel om vi ändrar värden som finns i StrWarm slider så kommer värdet som var 500 i början att ändras till det värdet vi bestämmer.   
Efter att vi har skickat och ändrar värden i databasen, så måste man kunna använda de för att kunna styra våran lampa. Och då kommer funktionen av mikrokontrollen. Mikrokontrollen kommunicerar också med Backend för att kunna få information från databasen. I mikrokontrollen så finns det två huvudfunktioner som gör att vi får en förändring i ljusstyrkan. GetfromDB (Get from databas) funktionen, och det den gör att den säger åt databasen att den vill hämta värden från databasen. Och i våran kod så har vi bestämt att vi vill hämta alla värden som finns i databasen. Och de värden vi hämtar vill vi använda som en output, alltså vi definierar StrWarm och StrCold som outputs, och vi sätter ett startvärden 0 på de outputs. För att kunna göra något vettigt med de värden vi hämtar från databasen med hjälp av Backend, så måste vi ha en Updatevalues funktion. Så när vi till exempel tänder lampan, så hämtas värden som vi skickar från våran Frontend när man trycker på On knappen, från databasen genom Backend, sedan uppdateras de värdena vi har när vi utför denna funktionen (I detta fallet så är värdena 0) till de nya värden vi får (Vilket i detta fall är 500). Sedan så sätter vi de två funktionerna i en Void Loop så att den körs automatisk, så varje gång de sker en förändring i databasen så hämta de värden genom Backend och sedan uppdateras de gamla värden som fanns i outputs i mikrokontrollen, och då sker i fysisk förändring i ljusstyrkan, eftersom till de outputs så finns det inputs som är inkopplade, alltså de som tar värden vi får, vilket är led stripparna.

## 5.3 GoogleHome Detaljbeskrivning

Vi hade även använt oss av röststyrning med hjälp av Google Home. Man ska kunna aktivera Google Home och sedan säga ”Tänd lampan” eller så enkelt som ”På” för att kunna tända lampan. Vad den gör är att den kör en Postfunktion som postar det exakta samma startvärde som om man vore tända på lampan med appen, alltså ska den först köra Delete för att det ska inte blir duplicerade värden. Man kan dock inte släcka av lampan, eftersom enligt Dialogflow så kan Google Home just köra Post och inte Patch eller Delete. Därför kan vi tyvärr inte släcka lampan med Google Home.

# 6 Kod Utdraga

Vi har en smart funktion på knappen på startsidan. Eftersom vi har använt oss av Post/Delete istället för Patch för att tända lampan så finns det risk att man råka gå ut från appen utan att trycka på av knappen i Screen2, när man då öppnar appen igen måste man trycka på knappen igen för att gå till Screen2, då blir det 2 likadana rader i databasen. Detta kan ge ett sämre resultat med lampan eftersom istället för att ändra på 2 värde (StrCold och StrWarm) måste den ändra på 4 värde, vilket kan påverka lampan att uppdatera segt. Vi hade hindrat detta problem genom att köra Delete före Post och Navigate i första knappen i Screen1. Då ser vi till att det inte finns något kvar i databasen före vi tänder på lampan igen. Dock har den en nackdel och det är att lampans ljusstyrka kommer då att återställa tillbaka till 500.

DeleteDataFromServer = () =>{ //Skapar en metod som tar värdena name och price så använder vi dem för att skicka till databasen

    console.log("bla")

fetch('http://iot.abbindustrigymnasium.se:3000/grupp2', { //Skickar värdena till databasen

    method: 'DELETE',    //Post betyder skicka

        headers: { // skickar med vilkoren

    'Accept': 'application/json',

'Content-Type': 'application/json',

        }

 }).then((response) => response.json()) //gör om den till json

    .then((responseJson) => {   // Showing response message coming from server after inserting records.

        console.log(responseJson); //Ser hela meddelandet från server //Skriver vilken produkt som blivit tillagd

    }).catch((error) => { //Fångar fel

        console.error(error);

    });

}

InsertDataToServer =() => {

    console.log("äöäöl");

    fetch('http://iot.abbindustrigymnasium.se:3000/grupp2/Str',{ //Bestämmer vart det nya värdet hamnar.

        method: 'POST', //Bestämmer att ett nytt värde skapas.

        headers: {

            'Accept': 'application/json',

            'Content-Type': 'application/json',

        }

    }).then((response) => response.json()).then(responseJSON => { //Gör om responsen till json-format.

        console.log(responseJSON); //Gör så att vi ser om värdet har skickats.

    }).catch((error)=>{

        console.log(error); //Meddelar vad som är fel.

    });

}

render() {

if (!this.props.visible) {

return false;

}

return(

<View style={styles.component}>

<View style={styles.layouts}>

    <View style={styles.layout1}>

        <View style={styles.itemcontainer1}>

            <View style={styles.itemcontainer1Inner}>

<TouchableOpacity

                                        onPress={() => { this.DeleteDataFromServer(); this.InsertDataToServer(); this.props.navigation.navigate('Screen2', {});}}

                                        //Om man trycker på knappen kommer man till den andra skärmen där man kan ändra ljusets värme och ljusstryka. Samtidigt skapas ett nytt värde med 50% ljusstyrka.

                                >

                                        <Image

                                  source={require('../../img/OnOff.png')} style={styles.On}

                                />

                                    </TouchableOpacity>

            </View>

        </View>

    </View>

</View>

</View>

);

}

}

Frontend Koden

router.post('/', (req, res, next) => { //post funktion för att kunna tända lampan

var Createdlight= function(){

return new Promise(function(resolve,reject){

con.query('INSERT INTO `light` (`lightName`, `StrCold`, `StrWarm`) VALUES ("lightName", 500, 500); ', function (error, results) { //post i lightName "lightName" och 500 på både StrWarm och -Cold

if (error)

return reject (error);

else

return resolve(results)

});

})

}

Createdlight().then( Thelight => { //om man lyckades posta, ska det skicka ett meddelande

res.status(201).json({

message:"Success, new value!",

light: Thelight

})

}).catch(error => { //annars påminner det att blev nåt fel

res.status(500).json({

error: error

})

});

});

router.delete('/', (req, res, next) => { //delete funktion, delete allt från databas för att kunna släcka lampan

console.log(req.params.lightName);

var deleteRows = function(){

return new Promise(function(resolve, reject){

const lightName = req.params.lightName;

con.query('DELETE FROM light',[lightName], function (error, results) {

console.log(error);

if (error)

return reject(error);

else

return resolve(results)

});

})

}

deleteRows().then(result => { //skicka ett medddelande för att visa att den funkar

if (result.length == 0) {

res.status(404).json({

message: "Lights off"

});

}

else

res.status(200).json(result)

}) .catch(error => { //informera om det blir något fel

res.status(500).json({

error: error

})

});

});

router.patch('/', (req, res) => { //patch funktion för att kunna ändra ljus styrkan

const light = {

lightName: req.body.lightName,

StrCold: req.body.StrCold,

StrWarm: req.body.StrWarm

}

var updateValue = function(){

return new Promise(function(resolve, reject){

con.query('UPDATE `light` SET `StrCold` = ?, `StrWarm` = ? WHERE `lightName`= lightName', [light.StrCold, light.StrWarm, light.lightName], function (error, results) {

if (error)

return reject(error)

else

return resolve(results)

});

})

}

updateValue().then(result => { //informera

if (result.affectedRows > 0)

res.status(200).json(result);

else

res.status(404).json({

message: "Not found"

});

Backend

# 7 Utvärdering

## 7.1 Positiva sidor

* Det som vi tycker var bra med vårat arbete är, att vi hade en användbar funktion som

faktisk löser ett problem som man kan möta på kontoret när det kommer till belysningen.

* Hur våran app ser ut, är också något vi är ganska stolta över.
* Att vi använde GoogleHome som hjälpmedel i vårat projekt.

## 7.2 Utvecklingsmöjligheter

Om vi skulle har haft mer tid så skulle vi:

* Jobba mer på designen på våran armatur
* Vi skulle ha byggt en bättre mekanism som är mer anpassad för lampan, än projektordukens mekanism.
* Vi skulle ha använt mer sensorer än bara GoogleHome.

Något som kan vi borde ha gjort bättre är:

* CSS:en på våran app. Vi använde mycket (Margin och padding) funktionerna så att det inte gick att använda appen på vilken mobil som helst
* Våran planering skulle ha kunnat bli bättre om vi ha delat så att varje en får en helt egen del. Det vi gjorde var att vi delade Backen, Frontend, Arduino och Armaturen till tre olika delar. Det sättet är mer lärorikt men, inte tidsoptimerad.

## 7.3 Lampans framtid

Vi tycker att våran lampa kan bli mycket användbar i skolor och i kontorsarbete, där dator är mycket användbar, och belysningen spelarroll för hur optimalt man kan jobba, och spelar också roll för ögonen hälsa. Men vi måste först hitta på en bättre lösning än projektordukens mekanism. Det behövs något som är mycket billigare, och något som funkar mer optimalt, utan några krångel. En till viktigt perspektiv som vi måste tänka på om det ska tillverkas mer av våran lampa är, att vi måste använda mindre plast och icke miljövänliga, och ersätta med något som är mer miljövänlig, utan det spelar roll med lampans huvudfunktioner.

1. https://akriform.se/nyheter/vad-ar-akryl-och-hur-fungerar-det-fler-kommersiella-akrylprojekt/ [↑](#footnote-ref-2)