

Presentasjon

Tittel:	XXXXXX Selfservice & Reporter Tool
Oppgave:	Utvikle et verktøy som gjør det mulig for kundene å løse sine avspillingsproblemer uten å måtte vente på svar fra kundeservice. Verktøyet skal også kunne utføre refusjon om feilen ikke løses. Videre skal verktøyet automatisk lagre informasjon om klienten hver gang den brukes og ansatte skal kunne hente ut data fra denne tabellen samt laste opp nye løsningsforslag (som vises i verktøyet kundene bruker).
Periode:	3. januar til 28. mai
Gruppede medlemmer:	Cristhian Gertner, Henrik André Karlsen
Prosjektgruppe:	38
Veileder:	Thor Hasle
Oppdragsgiver:	YYYYYY XXXXXX AS
Kontaktperson:	Kristen Tinius Torp,+47 975 77 186

Sammendrag

Prosjektet skal gjennomføres som hovedprosjekt ved HIOA, avd. for ingeniørutdanning, i samarbeid med XXXXXX. Kort oppsummert består oppgaven i å utvikle et analyseverktøy som kundene kan bruke, med en administrasjonsdel som administreres av XXXXXX. Vi skal i hovedsak prøve å utvikle analyseverktøyet m/administrasjonsdel i PHP med innflytelse av andre klientsidespråk som JavaScript (JavaScript, AJAX, JSON, jQuery) og/eller ordinær Java med mulighet for flere språk. Tanken er at verktøyet skal kunne brukes av samtlige klienter som kan brukes på nettsiden www.XXXXXX.com.

Om bedriften

XXXXXX leverer filmleietjenester på nett med intensjon om å utvide sortimentet sitt til andre områder også, f.eks SMS-tjenester osv. XXXXXX ble startet opp våren 2011 og har bl.a vunnet prisen for “God design” i 2012 som deles ut hvert år av Norsk Designråd. XXXXXX har avdelinger i Oslo (YYYYYY bygget på Fornebu) samt i Trondheim. Høsten 2012 startet XXXXXX også å levere filmleietjenesten i Sverige og Danmark.

Dagens situasjon

XXXXXX supporterer p.t sine kunder kun pr e-post i tidsrommet 16.00-22.00, noe som ikke alltid fungerer like optimalt. Når kunder leier en film ønsker de som oftest ønsker å kunne se filmen umiddelbart. Dersom filmen ikke fungerer må de sende en e-post, og denne blir som oftest besvart i løpet av få minutter. Mange kunder glemmer imidlertid å sjekke e-posten for svar, og får dermed ikke løst problemet før neste dag. Da har de kanskje ikke tid til å få sett filmen likevel, selv om denne er tilgjengelig i ytterligere 24 timer. Supportløsningen fungerer også uheldigvis for kunder som opplever problemer tidlig på dagen, før supporttjenesten er åpen. Disse vil naturlig nok ikke motta svar før på ettermiddag/kveld, ofte mange timer etter at problemet oppsto.

Med det verktøyet vi ønsker å lage i oppgaven, vil man kunne fange opp en stor del av de kundene som har problemer med å spille av innhold og dermed kunne øke brukeropplevelsen for disse kundene.

Videre vil XXXXXX med et slikt verktøy kunne hente ut statistikk fra spesifikke dager/uker/måneder.

F.eks vil man kunne se at i uke 40, da det kom en ny versjon av Silverlight, var det mange kunder som opplevde feil. Da kan utviklerne se at siste Silverlightversjon ikke lenger fungerer like optimalt med playeren på nettsiden og kan dermed utføre de nødvendige endringene som skal til for å fikse dette.

Kundeserviceavdelingen kan på sin side, v.h.a administrasjonspanelet, laste opp nye løsningsforslag og avpublisere gamle løsningsforslag som ikke lenger er aktuelle for tjenesten.

Mål og rammebetingelser

Målet med oppgaven er å utvikle et verktøy som gjør det mulig for kundene å løse sine avspillingsproblemer uten å måtte vente på svar fra kundeservice.

Det er ønskelig at verktøyet skal kunne:

- utføre refusjon om feilen ikke løses.
- automatisk lagre informasjon om klienten hver gang den brukes
- la kunden laste ned et lite program som sletter Silverlight sin DRM-nøkkkel. Dette pga avspilling av DRM-beskyttet innhold blir umulig om denne filen er korrumpert.

Verktøyet bør selvsagt være så lett å bruke som overhodet mulig da kundene kan være alt fra ungdommer til eldre folk som vil se på såpeoperaer på XXXXXX TV.

Administrasjonsdelen skal kunne:

- la de ansatte hente ut statistikk som verktøyet har lagret hver gang noen har brukt det
- la de ansatte publisere nye løsningsforslag men også avpublisere gamle løsningsforslag der disse ikke lenger er aktuelle
- la de ansatte laste opp nye batcher med X antall hundre/tusen koder som brukes i forbindelse med refusjon

Løsninger

Vi ser for oss at leveransen blir skrevet i PHP med innflytelse fra klientsidespråk som JavaScript, AJAX, JSON, jQuery og muligens en liten Java-klient (eller et annet universelt språk) som kan gå inn og slette nevnte problematiske DRM-filer som tilhører Silverlight. Vi ser også for oss at vi med et eller flere universelle klientsidespråk vil kunne hente ut hardware informasjon om den enkelte maskinoppsett. (I forbindelse med hvilke data som sendes til XXXXXX for å kunne forbedre produktet).

Med universelle språk mener vi språk som minimum MacOS og Windows forstår. Fordi XXXXXX idag baserer seg på å levere tjenesten ved bruk av Silverlight, vil vi i denne omgang ikke lage noen feilsøkningsverktøy for iOS- eller Android-enheter.

Programmeringsspråk: PHP, JavaScript, AJAX, JSON, jQuery Java 1.7

Utviklingsprogrammer: Netbeans IDE 7.0.1, Eclipse Indigo

Database: MySQL

Diverse andre programmer: Paint.NET, FileZilla (FTP), m.f

Analyse av løsninger

Vi vurderte å skrive leveransen i .NET rammeverket, men falt av ulike årsaker isteden ned på PHP. Vi har kun hatt ett semester på oss til å både bruke og lære C# .NET, vi opplever det derfor som risikabelt å levere noe i .NET rammeverket dersom vi plutselig skulle møte på problemer som tar lang tid å rette opp i.

Hadde gruppen bestått av flere enn to personer, ville vi selvsagt hatt større grunn til å velge .NET rammeverket da tre hoder tenker bedre enn to og fallhøyden dermed ville blitt mindre. Med kun to personer, vil det bli kritisk ved en eventuell sykdom, skulle det vise seg at den gjenværende parten ikke kan rammeverket godt nok.

Fordi vi har jobbet med PHP rammeverket siden andre semester, føler vi oss dermed relativt sikre på at vi kan håndtere store problemer på kort tid, selv om en av oss skulle bli langtidssykemeldt.

Begge to har, som nevnt før, veldig god og positiv erfaring med PHP rammeverket, og vi føler oss rimelig sikre på at vi vil kunne gi brukere en like fin brukeropplevelse ved bruk av PHP-rammeverket som med .NET.

Nettsideanimasjoner og lignende gjøres uansett av jQuery samt noe CSS og man blir dermed ikke avhengig av å måtte bruke .NET sitt grafikkrammeverk.

Konklusjon

Verktøyet vil kunne være til stor hjelp for XXXXXXs kunder som vil kunne få problemet sitt løst der og da uten å måtte involvere kundeservice. Med en stadig voksende kundemasse, og ikke minst, ekspansjon til andre markeder, ser vi flere positive effekter av verktøyet:

- Kunder får øyeblikkelig hjelp
- Kundetilfredsheten øker
- XXXXXX kan hente ut mer nøyaktig statistikk og med dette dermed vite hvilke løsningsforslag de skal publisere eller rette opp i (om informasjonen ikke er korrekt).
- XXXXXX kan med letthet publisere nye løsningsforslag
- Utviklere kan se om nyere versjoner av f.eks Silverlight, Firefox, Chrome m.m skaper flere problemer når de blir lansert.
- XXXXXX slipper å måtte gå til skrittet å få telefonsupport fordi verktøyet tar unna store deler av kundemassen som ellers ville sendt e-post inn til kundeservice.

Vedlegg (som egne filer)

- Use case diagrammer for henholdsvis kunde og ansatt
- En foreløpig fremdriftsplan