**چک لیست استخراج شده از اسلایدهای درس:**

**ریسکهای تکنیکی:**

از آنجایی که ریسکهای تکنیکی در این فاز اهمیت زیادی دارند و باید در نظر گرفته شوند با جزییات این ریسکها نوشته شده است و بررسی شده که زبان و فریم ورک پیاده سازی تا چه حد حایز اهمیت است. به طور کلی می توان ریسکهای تکنیکی مربوط به افراد و تکنولوژی دانست که در محصول ارایه شده اطلاعات مربوط به آنها گزارش شده است.

**Architecturally significant requirement:**

مجموعه ای از نیازها و ریسکهای پروژه که می تواند محدود کننده ای برای تکنولوژی مورد استفاده به حساب بیاید. این مجموعه نیازها موجب می شوند تا با طراحی و پیاده سازی مختصر معماری در این بخش از مرتفع شدن این خطرات مطمین شویم. این نیازها همانطور که مشاهده می شود از عمده ترین نیازهای سیستم و به نوعی از اهداف اصلی سیستم هستند که به خوبی تشخیص داده شده و بیان شده اند.

**Usecase Realization :**

در این بخش تنها به پیاده سازی Activity Diagram برای هر کدام از Use case هایی که به این نمودار نیاز دارند، محدود می شود، نمودار های فعالیت با نام مربوط به مورد کاربرد مربوطه مشخص شده اند.

**نمونه‌ی اولیه‌ی واسط کاربری**

واسط کاربری در این مرحله، نیازی نیست که دقیقن و به طور کامل همان واسط کاربری نهایی باشد و می تواند تا حدی متفاوت با آن باشد، چراکه هنوز تکنولوژی های مورد استفاده در حال مشخص شدن هستند، به همین منظور با استفاده از نرم افزارهای تولید prototype، یک نمونه اولیه از ظاهر نرم افزار ارایه شده، که ظاهر پنجره ای آن و محل قرار گیری کامپوننت ها در آن تا حد زیادی با خروجی نهایی مطابق خواهد بود. همچنین در این واسط نحوه انتقال بین پنجره ها و چگونگی پر کردن فیلدها و درخواست ها را می توان مشاهده کرد. هدف از این محصول، ارایه اولیه به کاربر به منظور دریافت حس مشترک از کل سامانه و گرفتن تاییدیه محصول نهایی در ابتدای کار است و در صورتی که مغایرتی با فضای فکری مشتری از سامانه داشته باشد، باید در مرحله بعد اصلاح شود.

**کارتهای CRC :**

کلاس بندی اولیه مورد نظر برای سیستم، مسئولیتها و همکاران در این کارتها نشان داده شده است. همانطور که در مفاهیم شی گرا مطرح است، تلاش بر این است تا داده های مربوط به اعمال آنها در یک کلاس قرار بگیرد، به عنوان مثال می توانستیم، داده های مربوط به منبع مورد استفاده هر پروژه را در کلاس پروژه با جزییات ذخیره کنیم ولی این داده ها در کلاس منبع موجودند و انتساب آنها به پروژه تنها اطلاعی است که به نحوی در کلاس پروژه قرار می گیرد. همچنین در این کارتها سعی شده از ایجاد کلاسهای بزرگ و یا کلاسهایی که تنها حاوی داده هستند خودداری شود. کلاسها با توجه به مفاهیم refactoring ایجاد شده اند و در صورتی که کلاسی نقش زایدی مثل اختصاص منبع به پروژه و یا از این قبیل واسطه گری ها را داشته حذف شده و تنها کلاسهای مهم که نقش اساسی در سیستم دارند و در صورت حذف آنها کلاس دیگری به سادگی نمی تواند اعمال آنها را انجام دهد، باقی مانده اند.

**نمودارهای فعالیت:**

نمودارهای فعالیت مربوط به موارد کاربرد رسم شده است، برای موارد کاربردی که تکراری بوده اند، مثل موارد کاربردی که برای مرتب سازی بر اساس معیارهای متفاوت بود، تنها یک نمودار فعالیت رسم شده و از تکرار زاید نمودارها خودداری شده است. در هر نمودار فعالیت سعی شده تا پیش شرطها و اتفاقات ممکن در طول پروسه فعالیت نیز در نظر گرفته شود و بخش هایی که داده یا شی منتقل می کنند با مربع مخصوصی مشخص شده است. نمودار های فعالیت به عنوان یکی از محصولات این فاز هنوز به کمال نرسیده و در فازهای بعد تکمیل می شود.

**Executable architectural baseline :**

معماری قابل اجرا در این بخش با توجه به توضیحات داده شده نیاز نبود و تمام ریسکها و نیازمندی هایی که منجر به نیاز به این معماری بودند (همانطور که در توضیحات مربوطه گفته شده) مرتفع شدند.

**زمان بندی:**

زمان بندی و اختصاص کارها به اعضای گروه با توجه به فاز قبل و تغییرات ددلاین ها تغییر کرد و به عنوان خروجی این فاز در نظر گرفته شده است. این زمان بندی تا حدودی، مشغله اعضای گروه و تمایل انجام کار را با توجه به زمان در آنها بررسی می کند.